LA CAO SANS PEINE

Jean-Pierre Petit

PROLOGUE :

"La CAO sans peine", livre paru initialement en 1984, sous le titre PANGRAPHE (la maison d'édition a disparu depuis longtemps) est une initiation à la conception assistée par ordinateur. A l'appui, un programme écrit en BASIC, qui tournait à cette époque sur un Apple II, dont la mémoire centrale était de 48 K (près de mille fois inférieure à la mémoire de mon PC). Le BASIC est un peu tombé en désuétude, pour des raisons de mode. En effet des BASIC plus récents, s'affranchissant des défaut du langage primitif (suppression des numéros de lignes, emploi d'étiquettes pour repérer les sous-programme et les adresses en général, redéfinition des variables, introduction d'un compilateur) a fini par égaler en performances d'autres langages considérés comme plus évolués, comme le PASCAL.

Mais le BASIC avait un avantage, il était très simple à apprendre, à enseigner. Le lecteur-programmeur n'aura guère de difficultés à adapter le logiciel présenté ici dans un autre langage de son choix. L'essentiel est donné : l'architecture des données, les manipulations de base (transformations géométriques), la façon de créer des images à volonté et de créer des vues en perspective.

PANGRAPHE fut suivi par deux autres logiciels, toujours écrits en BASIC : SUPERAMSTRAD-3D et SCREEN, où les parties cachées étaient éliminées en utilisant "l'algorithme du peintre".

PANGRAPHE servit de point de départ à plusieurs équipes de programmeurs qui développèrent des produits. Certains eurent même la gentillesse de me dédier leur logiciel, évidemment beaucoup plus perfectionnés. Pas mal de gens s'amusèrent beaucoup avec SUPERAMSTRAD-3D, qui tournait sur un "6128", autre antiquité du bestiaire informatique. Avant la lettre il s'agissait d'une "programmation orientée objet", avec recours à une "mémoire virtuelle". Le lecteur pouvait disposer d'une bibliothèque d'objet et réaliser tes tas de nouvelles opérations, comme des fusions d'objets. Les "translations-fusions" permettaient de multiplier des fenêtres sur la facade d'un immeuble, ou des colonnes sur celle d'un temple. Les translation-rotation-fusions combinées permettaient de créer des escaliers en colimaçon à partir d'une simple marche, etc.

La CAO est un monde infini. A l'époque nous étions limités par les mémoires et les vitesses de nos machines, non par notre imagination. Au début des années quatre-vingt, certains se souviendront peut être que j'avais présenté, à TF1, un petit film créé par un Apple IIe, où l'ont voyait une

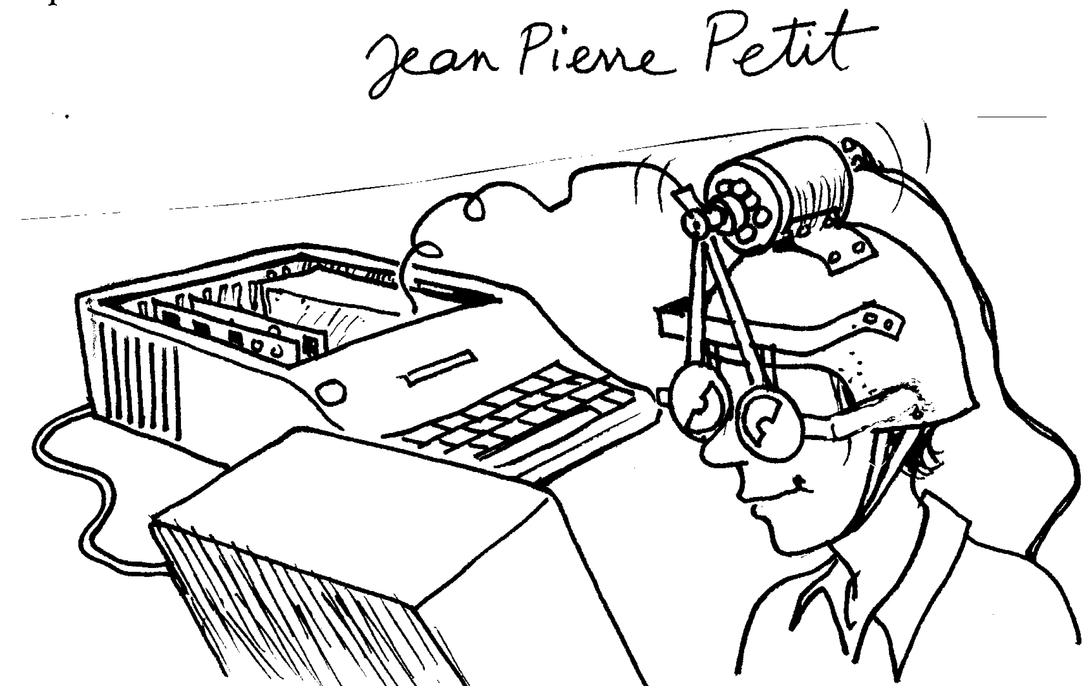
succession de vues d'avion d'un village, parties cachées éliminées. Les spécialistes se demandèrent comment une machine aussi lente, cadencée à 2 mégahertz, pouvait produire des images aussi complexes, à une telle vitesse. En fait, elles avaient été "précalculées" et stockées sur une carte mémoire additionnelle de 512 K (les "pages écran" Apple faisait... 8K). Bref ceci préfigurait, avant la lettre, le ... videodisque.

A l'époque où je dirigeais le Centre Informatique d'Enseignement que j'avais créé à la Faculté des Lettres d'Aix-en-Provence nous avions utilisé la capacité de faire afficher par l'Apple II deux "pages écran", en alternance, à un rythme élevé, pour créer un système de vision en relief. Un moteur électrique, fixé sur un casque de moto, actionnait des caches rotatifs, masquant et découvrant les yeux droit et gauche de l'utilisateur. Un contacteur, délivrant un signal en 5 volts sur l'une des entrées de la machine, commutait les pages écran "œil droit" et "œil gauche", de manière synchrone, l'ensemble ayant été baptisé "STEREOCYCLETTE".

Ma foi, si le lecteur de la "CAO sans peine" manifestent leur curiosité, nous pourrons leur expliquer encore beaucoup de choses. Rien ne ressemble plus à la CAO qu'un ensemble de boites de MECCANO. Dans la mesure où le "source" est fourni (ce qui fut toujours le cas) rien n'empêche un nouveau venu d'adjoindre quelque gadget de son cru, un sous programme permettant d'engendrer toutes les coquilles d'escargot possibles, par exemple.

La CAO, les programmes de pilotages ou de jeu d'échec, sont maintenant largement répandus et de plus en plus performants, comme les voitures télécommandées à pile, où les robots qui marchent. Programmer est devenu dépassé ou considéré comme une affaire de pros. Pourtant vous ne trouverez nulle part un logciel assez fou pour vous permettre, comme PANGRAPHE, de voir ce que vous avez derrière la tête.

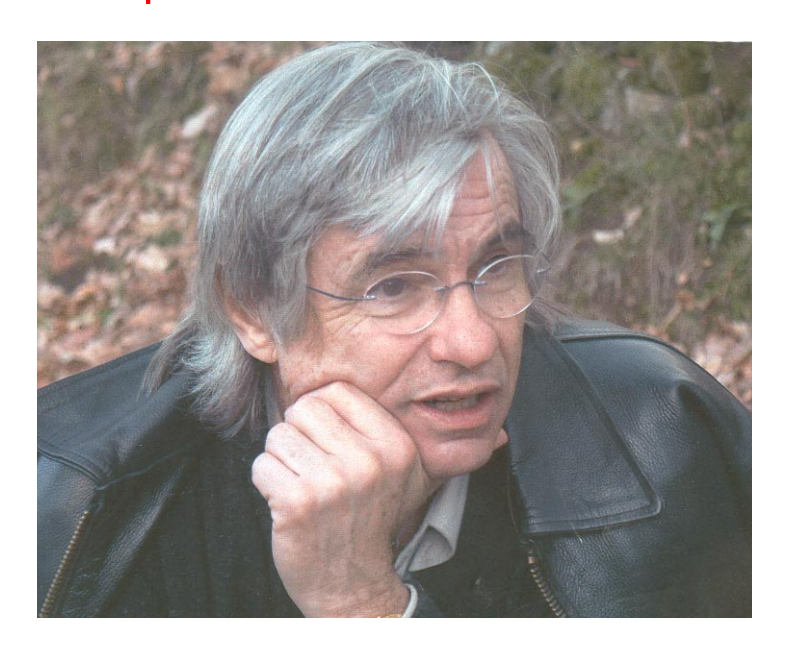
Jean Pierre Petit



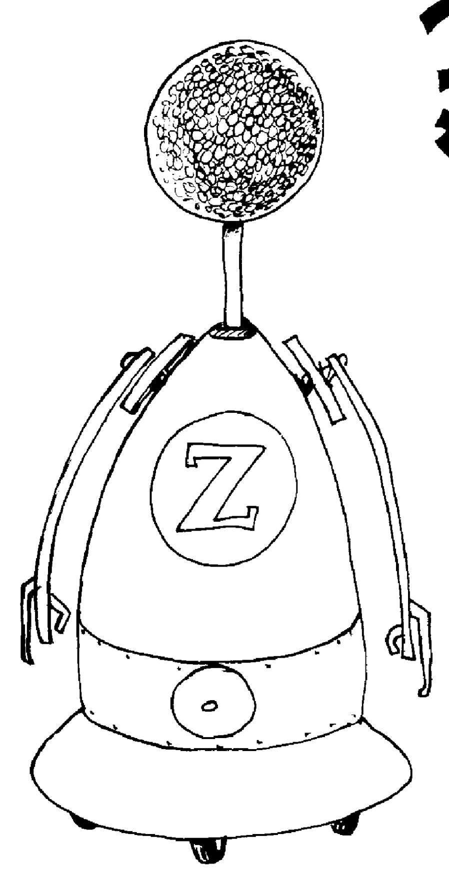
Savoir sans Frontières

Association Loi de 1901
Villa Jean-Christophe, 206 Chemin de la Montagnère, 84120 France

http://www.savoir-sans-frontieres.com



Jean-Pierre Petit, Président de l'Association : Ancien Directeur de Recherche au Cnrs, astrophysicien, créateur d'un genre nouveau : la Bande Dessinée Scientifique. En 2005 il a décidé de mettre ses ouvrages au nombre d'une vingtaine, dans le domaine public en les rendant téléchargeables gratuitement à partir de son site. Il a également créé l'association Savoir sans Frontières qui s'est donnée pour but de distribuer gratuitement le savoir, y compris le savoir scientifique et technique à travers le monde. L'association, qui fonctionne grâce à des dons, rétribue des traducteurs à hauteur de 150 euros (en 2006) en prenant à sa charge les frais d'encaissement bancaire. De nombreux traducteurs accroissent chaque jour le nombre d'albums traduits (en 2005 en 18 langues, dont le Laotien et le Rwandais).
Le présent fichier pdf peut être librement dupliqué et reproduit, en tout ou en partie, utilisé par les enseignants dans leurs cours à conditions que ces opérations ne se prêtent pas à des activités lucratives. Il peut être mis dans les bibliothèques municipales, scolaires et universitaires, soit sous forme imprimée, soit dans des réseaux de type Intranet.
L'auteur a entrepris de compléter cette collection par des albums plus simples d'abord (niveau 12 ans). Egalement en cours d'élaboration : des albums « parlants » pour analphabètes et « bilingues » pour apprendre des langues à partir de sa langue d'origine.
L'association recherche sans cesse de nouveaux traducteurs vers des langues qui doivent être leur langue maternelle, possédant les compétences techniques qui les rendent aptes à produire de bonne traductions des albums abordés.
Les dons (chèque libellés à l'ordre de Savoir sans Frontières) sont également les bienvenus. Les ressources de l'association sont en 2006 principalement affectées aux nouvelles traductions.



Bien des années se sont écoulées depuis le début de la **DEUXIÈME ÈRE**. Depuis qu'il n'y a plus d'hommes sur la Terre.
J'ai pensé qu'il était bien que l'un d'entre nous tente de mettre un peu d'ordre dans tout ce qui s'était passé.



Il a pu être établi de manière certaine que William Straton, dont voici une photo prise un an avant sa mort, fut l'homme qui réduisit à néant, en quelques mois, un travail qui avait pris des centaines de millions d'années.

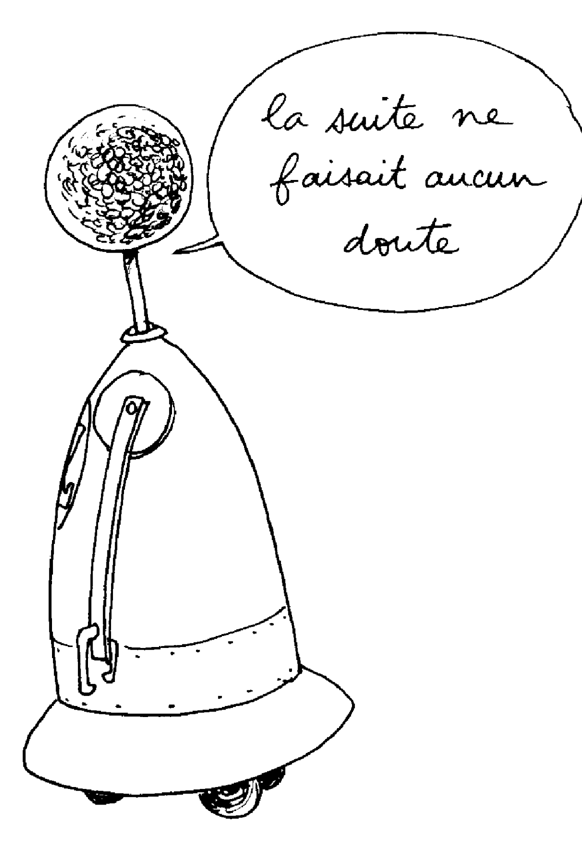
C'était un chercheur semblable à des milliers d'autres, qui avait un obscur emploi d'aide biologiste à l'International Cosmetics and Drugs Corporation.
William n'avait rien d'un docteur Mabuse. Il oeuvrait de manière routinière sur une souche virale icosaédrique VI-754, un bête virus de la grippe.

Cela vaut peut-être la peine de consacrer quelques lignes à cet accident qui effaça un patient travail de conquête planétaire.

les hommes n'avaient pas trente six façons de lutter contre les grippes. Un bon coup de fièvre, et tout était réglé, les virus ne supportant pas cette forte température.

Par le plus grand des hasard, il semble que Straton ait, à la suite de manipulations génétiques totalement anarchiques, créé une souche virale **THERMORÉSISTANTE**.







le chat de Stratton apporta peut-être la petite pichenette au destin. Peu importe... William attrapa le premier cette fichue grippe.

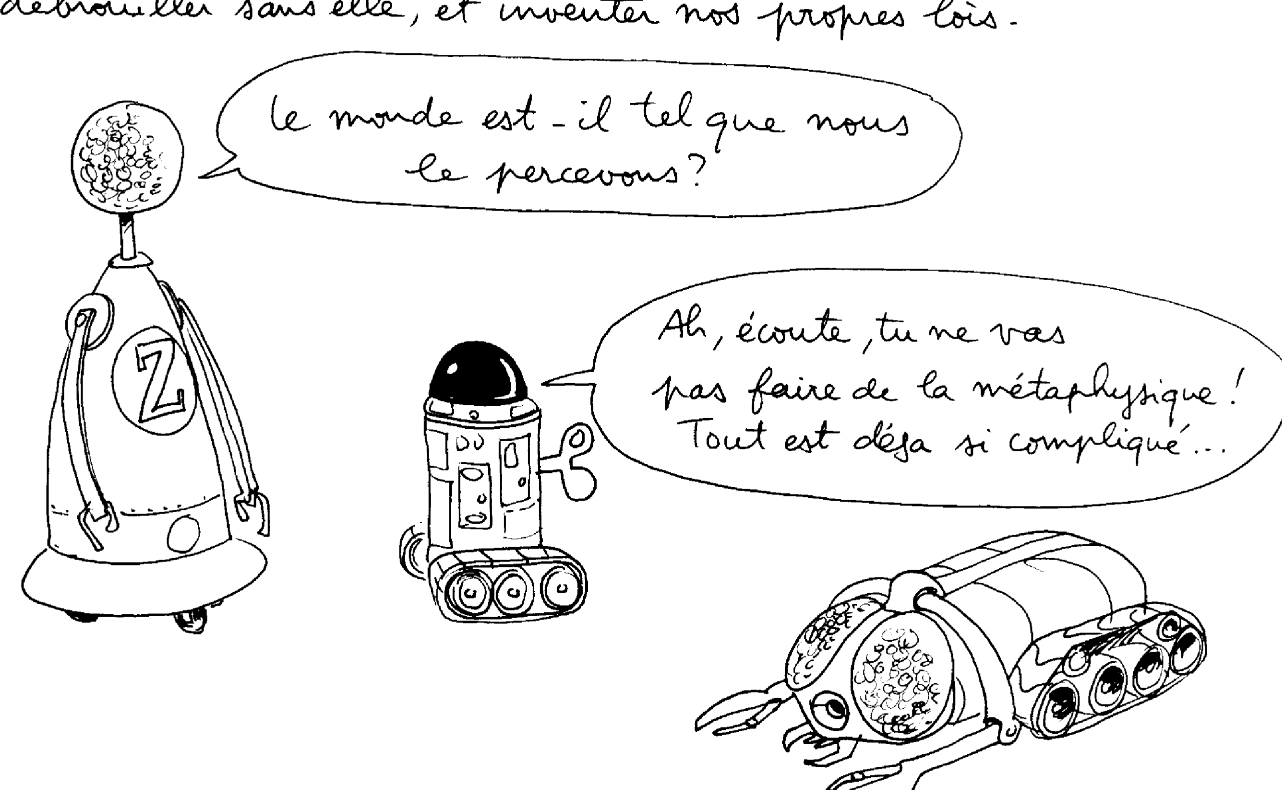
Sa température monta, monta…
Quand elle atteignit quarante deux degrés, il passa de vie à trépas.

La logeuse de Stratton fut la seconde victime. Puis les habitants de la rue Texington… ceux de Londres… puis l'Europe, le monde entier, les humains, y passèrent jusqu'au dernier.

Et nous restâmes seuls, tous seuls, pour gérer cette fichue planète. Heureusement certains d'entre nous étaient devenus intelligents. Enfin… un minimum pour que tout ne parte pas à vau l'eau.

Nous mîmes nos savoirs en commun. Et depuis nous évoluons comme nous pouvons.

Ne croyez pas que cela soit simple. Les êtres vivants, eux, n'ont qu'à suivre la nature. Nous, nous devons nous débrouiller sans elle, et inventer nos propres lois.



la conduite de la planète n'est pas un si grand problème. Ce sont les hommes qui en faisaient toute une histoire. Sur le plan technique, nous dominons assez bien la question. Mais il restent des choses bien obscures.

Tenez, par exemple, moi j'ai été nommé directeur du musée du Louvre. Ce local contient des centaines d'objets variés. Mais nous ne comprenons pas leur raison d'être en ce lieu.

Dans le temps, les hommes y venaient souvent. Ils regardaient par exemple les tableaux pendant de longues minutes. Nous avons essayé, mais cela ne nous donne absolument rien. En tout état de cause c'est beaucoup moins bon que des photos.



Il y a des statues auquelles il manque des bras, et parfois, la tête. En nous inspirant d'autres documents, nous avons un peu arrangé cela.

Dans ces grandes pièces vides, je m'ennuie un peu. J'ai tout mis en mémoire : les fresques, les sculptures, les tapisseries, et je ne vois pas quoi faire d'autre...

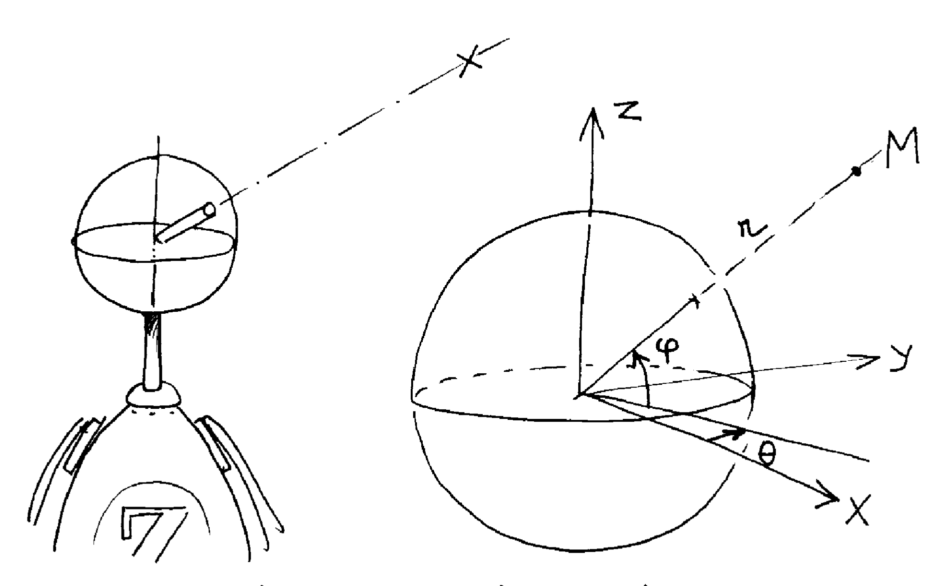
Un collègue, Bernie, a suggéré que nous pourrions peut être percer ce secret que les hommes n'ont pas pensé à nous laisser en nous quittant, en apprenant à **DESSINER.**

Je lui ai demandé ce qu'il entendait par là

Bernie m'a expliqué que les yeux des hommes ne fonctionnaient pas comme le nôtre.

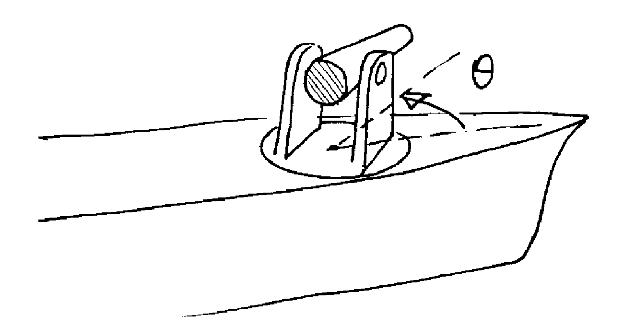
LE DESSIN

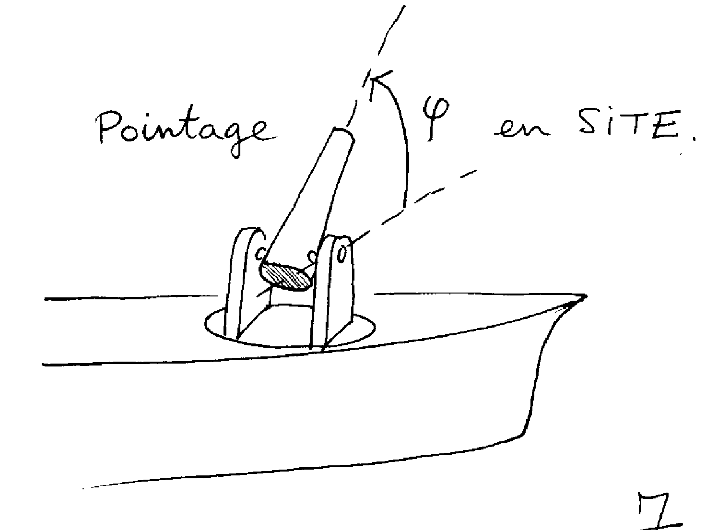
Bien avant le début de la deuxième ère, les hommes nous avaient déja configurés pour que nous percevions les objets, le monde sensible, en **COORDONNÉES SPHÉRIQUES**:



Notre oeil est en fait constitué d'une myriade de petits yeux élementaires, chacune de ces ocelles pointant dans une direction particulière.

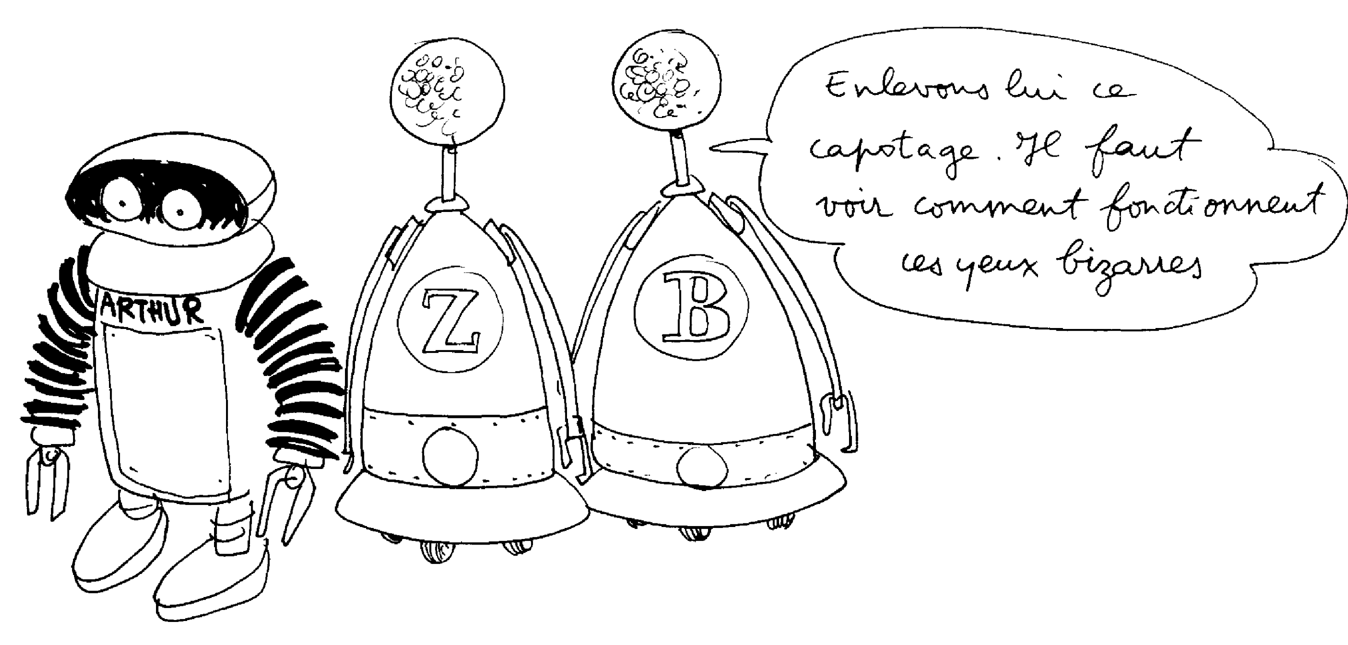
Deux angles, θ et φ, suffisent à repérer cette direction de pointage. Théta est l'**AZIMUT** et phi le **SITE**.

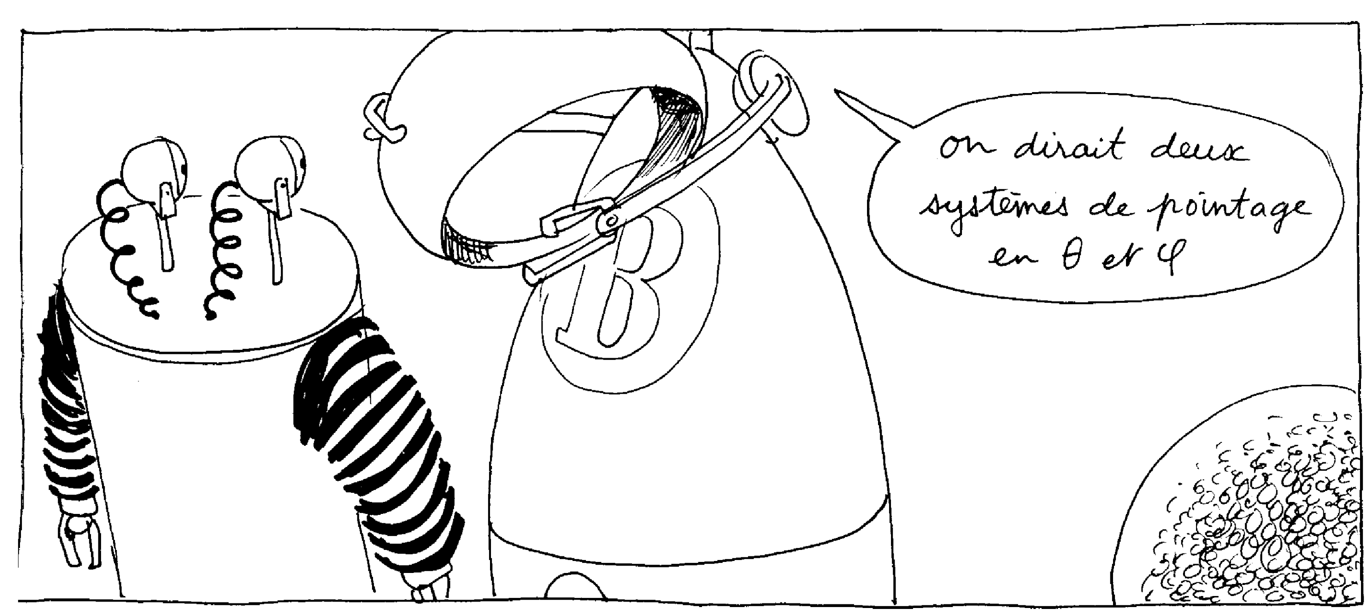




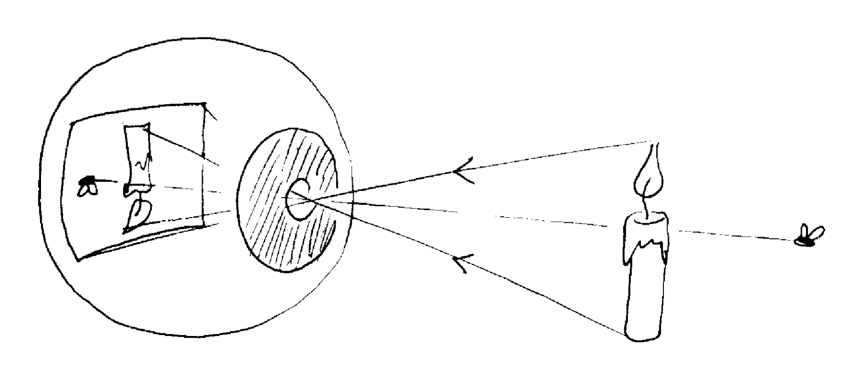
Chaque ocelle est, de plus, un radar minuscule, qui fournit la distance r à l'objet. Un seul oeil, sphérique, permet donc de connaître avec précision tout ce qui nous entoure, dans toutes les directions.

Bernie m'explique donc que les yeux des hommes, plus primitifs, n'étaient pas fichus comme cela. Nous avons déniché dans une remise un robot de l'ancienne génération, à vision binoculaire, construit sur le modèle de l'homme.





A la suite d'un examen plus poussé, on s'aperçoit que chacun de ces yeux permet de former l'image de tout objet extérieur sur un écran intérieur appelé rétine :



Ce qui n'est rien d'autre que la constitution, à partir d'un objet tridimensionnel, d'une image **PLANE**, se formant sur une **CALOTE SPHÉRIQUE** (le fond de l'oeil).

Dans l'opération **DESSIN**, on place tout simplement l'écran **DEVANT** l'oeil lui-même.

Ecran qui est une portion de sphère, sur laquelle les **POINTS-IMAGE** seront donc repérables à l'aide des deux coordonnées θ et φ (site et azimut).

Si l'écran est assez "petit" c'est à dire si l'**OUVERTURE ANGULAIRE**, les valeur maximales et minimales de θ et φ restent faibles ce fragment de calote sphérique sera très voisin d'un **PLAN.**

l'absence de mesure directe de la coordonnée radiale r représente une évidente et importante perte d'information. On se demande vraiment comment l'homme pouvait se débrouiller pour avoir une conception quelconque de l'espace dans lequel il vivait.

Bernie pense que cela devait aller avec un **SOFT**, c'est à dire des **STRUCTURES MENTALES**, très différentes. Et c'est peut-être la raison pour laquelle nous restons de marbre devant les peintures du musée du Louvre.

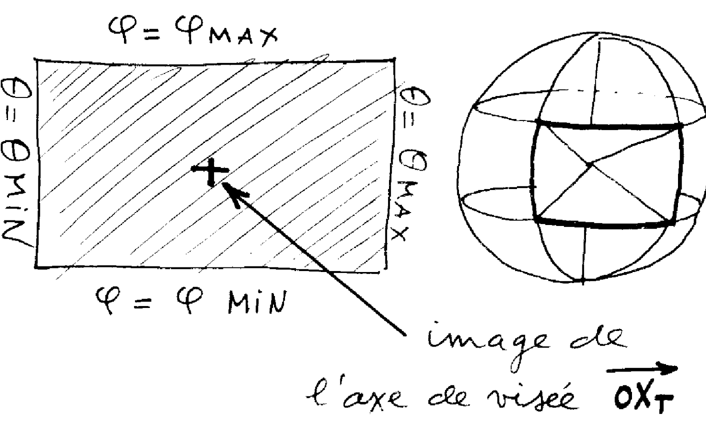
Il faut donc bien cerner ce problème du **DESSIN**, qui n'est rien d'autre qu'une opération de projection cônique, le sommet du cône étant l'oeil de l'observateur. On placera l'écran, sur la sphère de visée de telle manière que le centre de l'écran coïncide avec l'origine des coordonnées angulaires (c'est à dire que ce centre correspond à $\theta = 0$ et $\varphi = 0$) On appelera cet axe $\overrightarrow{OX_T}$

ŒIL O

X_T

les bords de notre écran sont constitués par des fragments de méridiens et de parallèles.

J'ai fini par comprendre que les humains utilisaient un système de repérage des **POINTS OBJETS** basé sur des **COORDONNÉES SCALAIRES.**

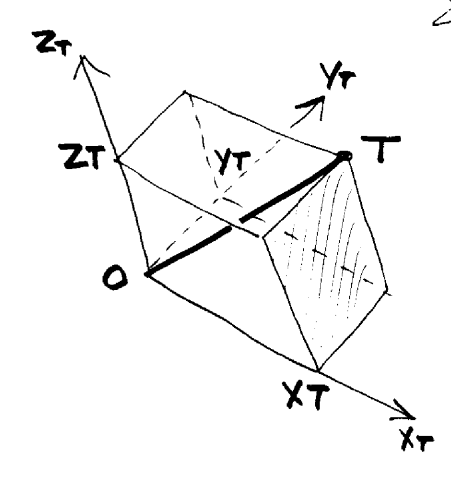


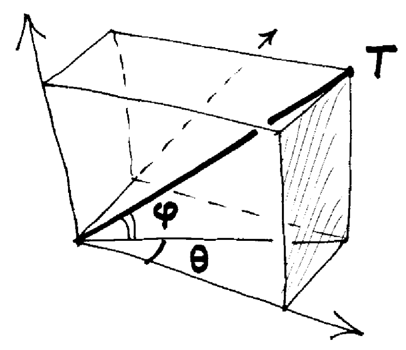
DESCRIPTION OBJET

L'Univers mental de l'homme est vraiment compliqué. Il semble qu'il projette, mentalement, le vecteur objet $\overrightarrow{OT}$, c'est à dire le segment joignant l'oeil et le **POINT OBJET T**, perpendiculairement, sur trois axes $\overrightarrow{OX_T}, \overrightarrow{OY_T}, \overrightarrow{OZ_T}$.

J'ai trouvé ces informations dans le soft graphique assez primitif du robot de l'ancienne génération. La lettre **T** me semble être une survivance du vocable anglo saxon TARGET, qui veut dire CIBLE. Les capteurs visuels ont dû être, en leur temps, des versions perfectionnées des systèmes de guidage des missiles AIR-AIR.

CLOK ?

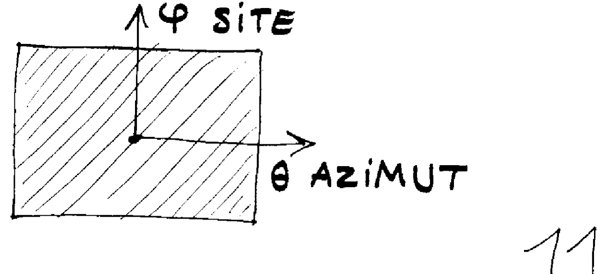




Tout **POINT OBJET T** correspond donc à un ensemble (**XT, YT, ZT**) de trois **COORDONNÉES CARTÉSIENNES** associées au **REPÈRE FIXE** $(\overrightarrow{OX_T}, \overrightarrow{OY_T}, \overrightarrow{OZ_T})$

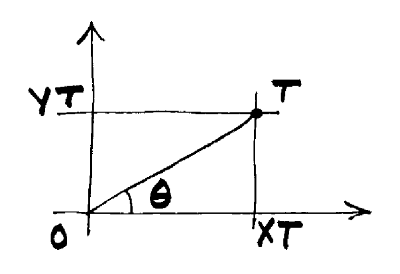
L'opération **DESSIN** consiste donc, à partir de la donnée des coordonnées cartésiennes (**XT, YT, ZT**), à recalculer les coordonnées "apparentes" que sont l'**AZIMUT θ** et le **SITE φ**.

Ceci fait, il suffira de reporter ce **POINT IMAGE** sur un écran



CALCUL AZIMUT & SITE

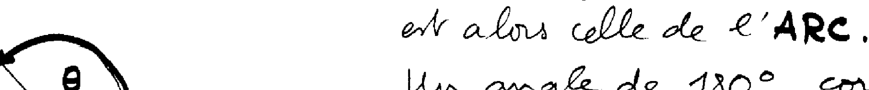
Commençons par l'**AZIMUT θ**. Le rapport **YT/XT** est, par définition la **TANGENTE** de l'**ANGLE θ**. C'est une valeur algébrique. Selon les signes de **XT** et de **YT**, cette tangente peut avoir des valeurs positives ou négatives. Détail important : si **XT=∅** la valeur de cette tangente est **INFINIE**.



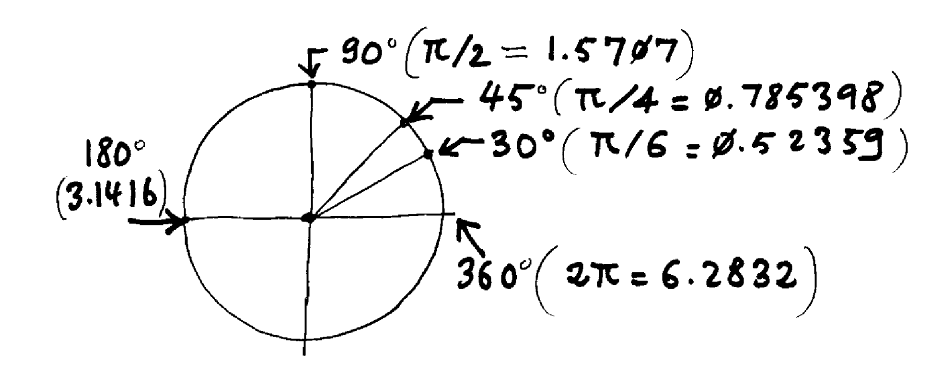
Un **SOUS-PROGRAMME**, matérialisé par l'instruction **TAN** permet de passer de la valeur de l'angle en radians à la valeur de la tangente.

Une instruction **ATN** réalise l'opération inverse. C'est à dire qu'à partir de la donnée du rapport **YT/XT**, elle permet de remonter à la valeur de l'angle. On appelle cet angle l'**ARCTANGENTE**.

Quand on mesure un **ANGLE** en **RADIANS**, on le repère à l'aide d'un cercle de **RAYON** égal à **1**. La mesure de l'angle est alors celle de l'**ARC**.



Un angle de 180°, correspondant à la demi circonférence, vaut donc **π** c'est à dire **3.1416** radians.



Un petit essai rapide :

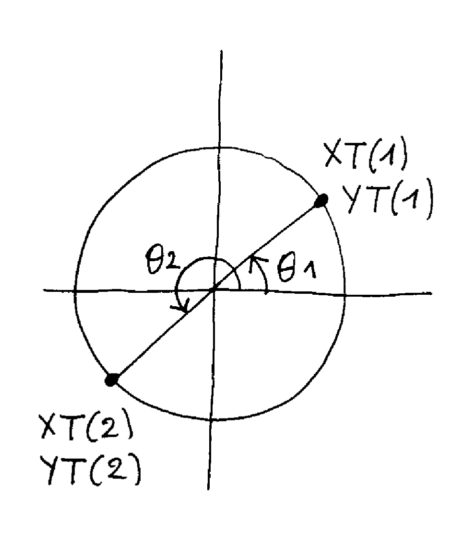
Quand l'angle vaut 45°, c'est à dire $\pi/4 = 0.785398$, XT et YT sont égaux, et la tangente vaut l'unité.

→ PRINTTAN(Ø.78539) donne 1

→ PRINT ATN(1) donne Ø.785398

ATTENTION !

L'instruction **ATN** n'est pas suffisante pour déterminer l'**AZIMUT θ**.



Soient deux points 1 et 2, diamétralement opposés, et de coordonnées (XT(1), YT(1)) (XT(2), YT(2))

On a :

XT(1) > Ø ; YT(1) > Ø ; (YT(1) / XT(1)) > Ø

XT(2) < Ø ; YT(2) < Ø ; (YT(2) / XT(2)) > Ø

YT(2) / XT(2) = YT(1) / XT(1)

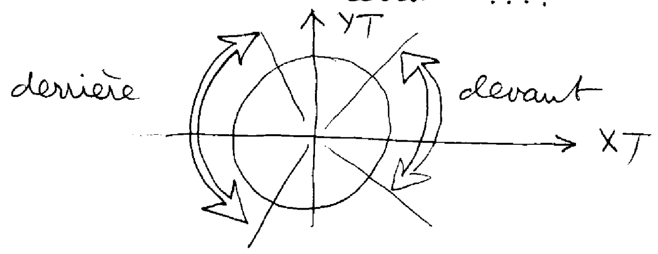
les deux valeurs de la **TANGENTE** de ces deux angles θ_1 et θ_2 sont identiques. Conséquence, pour ces deux points 1 et 2 les ordres PRINT (ATN (YT(1) / XT(1)))
et PRINT (ATN (YT(2) / XT(2)))
conduiront à la même valeur (en radians) de θ, égale à θ_1.
Il faut envisager un moyen de lever l'ambiguité.

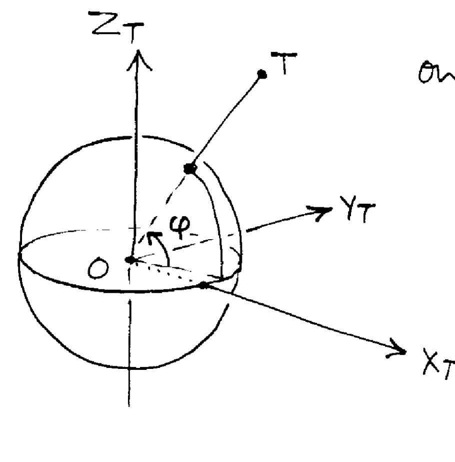
Si XT et YT sont négatifs, rajouter π à la valeur calculée à l'aide de l'instruction ATN

Si XT est négatif et YT positif l'instruction ATN fournira un angle θ' compté négativement. On aura $\theta = \pi + \theta'$ (puisque $\theta' < \emptyset$).

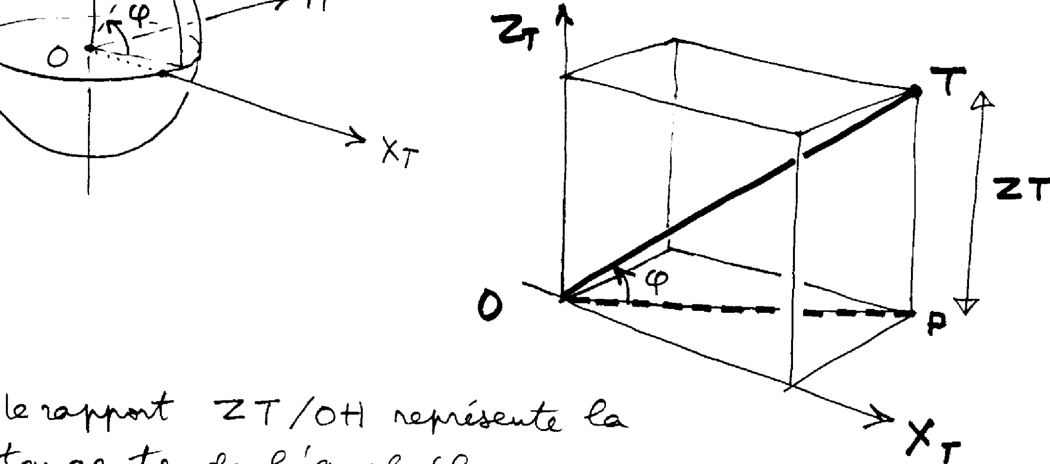
Dans les deux derniers quadrants, les valeurs de θ se calculent à l'aide de l'instruction ATN. Si XT et YT sont positifs, la valeur de l'arc tangente sera positive. Si XT est positif et YT négatif, la valeur de l'arc tangente sera négative

Bernie m'a fait remarquer que si on demandait à l'ordinateur de bord de construire l'image, sur un écran, d'un ensemble de points donnés par des coordonnées (XT, YT, ZT), et si on ne prenait pas cette précaution, celui-ci plaquerait sur une même image ce qui se trouverait devant et derrière sa tête ! Des valeurs de θ inférieures en valeur absolue à 1,57∅7 (90°) correspondent à des objets situés "devant". Les deux premières images expliquent la source de la confusion possible entre le "derrière" et le "devant"....

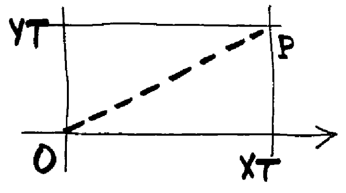




on a pas ce problème là avec le **SITE φ** puisque cet angle est compris entre $+\pi/2$ (zenith) et $-\pi/2$.



le rapport ZT/OH représente la tangente de l'angle φ.
Pour calculer OH nous utiliserons le Théorème de Pythagore

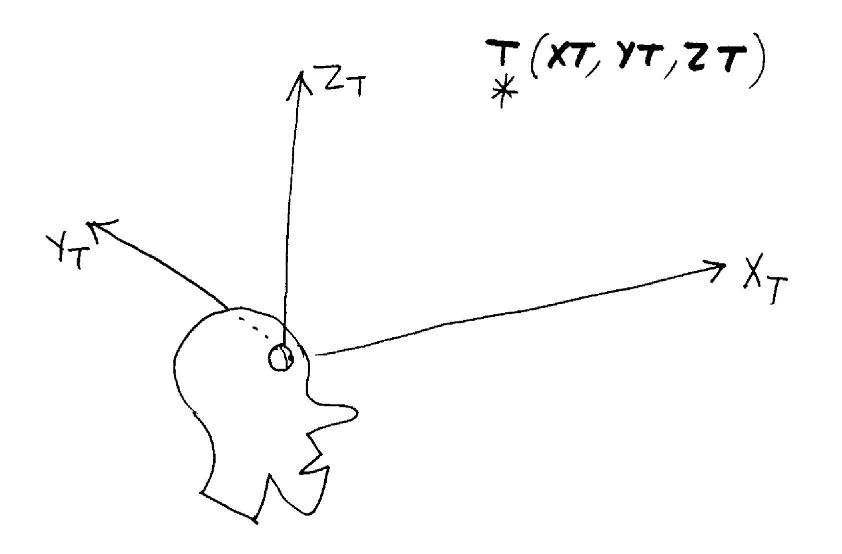


$OP^2 = XT^2 + YT^2$ ou $OP = \sqrt{XT^2 + YT^2}$

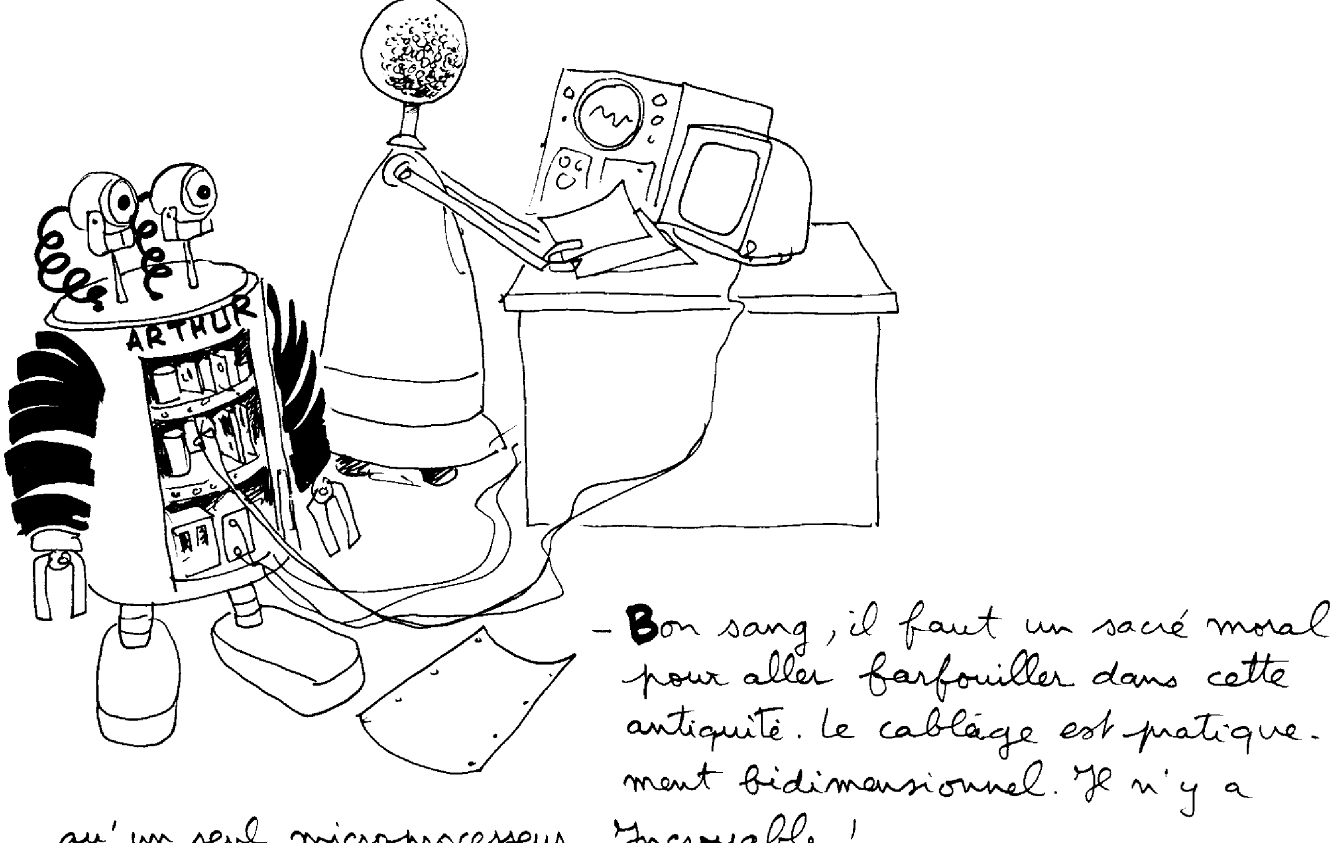
ou OP = SQR(XT*XT + YT*YT)

angle de site : S = ATN(ZT/OP)

En résumé, imaginons un sujet qui observerait un ciel empli d'étoiles. Chaque étoile étant un point objet de coordonnées XT, YT, ZT. Supposons ensuite que le sujet regarde dans la direction $\overrightarrow{OX_T}$ et que l'axe de sa tête coïncide avec $\overrightarrow{OZT}$



l'azimut et le site apparent de l'étoile correspondraient aux formules ci-dessus.



- **B**on sang, il faut un sacré moral pour aller farfouiller dans cette antiquité. Le cablage est pratiquement bidimensionnel. Il n'y a qu'un seul microprocesseur. Incroyable !
Si nous arrivons à réactiver ce bac, cela m'étonnerait qu'il arrive à aligner deux idées de suite...
Les mémoires sont structurées en couches parallèles, au lieu de se loger dans des cristaux de Titane, comme les nôtres. Et, tenez-vous bien, tout cela fonctionne à la température ambiante. Aucune supraconduction ! Ce qui fait que cet imbécile-là, s'il réfléchit un peu trop, doit se mettre à chauffer....
J'ai trouvé au musée des Sciences et des Techniques un très vieil ouvrage qui traite du **BASIC**. Une langue morte.
J'ai trouvé aussi un écran vidéo encore en état de marche. Une chance !

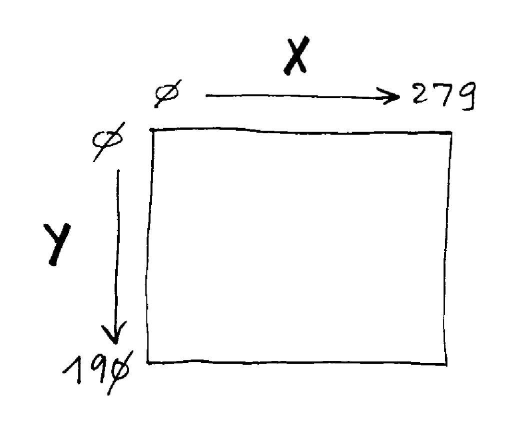
Et après bien des heures d'efforts, j'ai réussi à décrypter un des disques d'Arthur, qui se réfère au graphisme. Nous sommes dans la bonne voie.

Bernie trouve aussi cela assommant. Mais si, au bout du compte, ce travail nous permet d'éprouver nos premières impressions esthétiques, alors cela en vaut peut-être la peine.

Dans ce langage préhistorique, voici une suite d'instructions donnant l'azimut et le site à partir des coordonnées cartésiennes.

```
10 HOME : REM EFFACAGE ECRAN
20 INPUT"XT=";XT
30 INPUT"YT=";YT
40 INPUT"ZT=";ZT
50 IF XT=0 AND YT=0 ANDZT=0 THEN PRINT
"OEIL ET OBJET CONFONDUS":END
60 OP=SQR(XT*XT+YT*YT)
70 IF OP=0 AND ZT>0 THEN PRINT:PRINT"
SITE=+PI/2":PRINT:PRINT"AZIMUT INDETERMINE":END
80 IF OP=0 ANDZT<0 THEN PRINT:PRINT"
SITE=-PI/2":PRINT:PRINT"AZIMUT INDETERMINE":END
90 S=ATN(ZT/OP)
100 IF XT=0 ANDYT>0 THEN A=1.5708:GOTO 160
110 IF XT=0 ANDYT<0 THEN A=-1.5708:GOTO160
120 IF YT=0 AND XT<0 THEN A=3.1416
130 A=ATN(YT/XT)
140 IF XT<0 AND YT<0 THEN A=A+3.1416
150 IF XT<0 AND YT<0 THEN A=A-3.1416
160 A=A*180/3.1416:REM CONVERSION DEGRES
170 S=S*180/3.1416
200 PRINT"AZIMUT=";A:PRINT
210 PRINT"SITE=";S:GET A$:GOTO10
```

Il ne reste plus qu'à reporter cela sur un écran. Celui que j'ai trouvé a une capacité d'affichage de

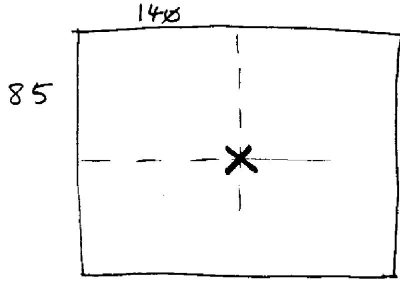


∅ à 279 sur X
∅ à 19∅ sur Y

X se "plote" de gauche à droite et Y de haut en bas ....
C'est comme cela ....

Bon, comment passer de ces valeurs en radians de l'azimut et du site à des "coordonnées écran" que je vais simplement appeler **X** et **Y** ?

Primo je situe l'axe optique au point (X = 14∅ ; Y = 85)



L'Ecran limite un certain **CHAMP VISUEL**, c'est une sorte de fenêtre.

AN
AN

Nous appelerons **OUVERTURE ANGULAIRE AN** le demi angle du champ visuel latéral.

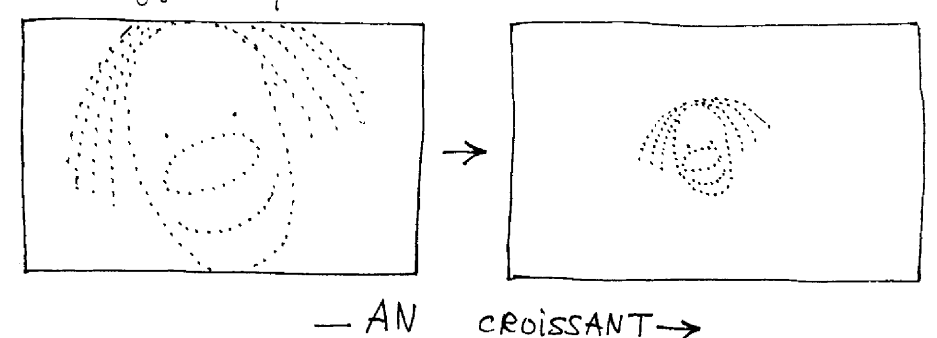
On peut se débrouiller pour que le point soit sur le bord droit ou gauche de l'écran lorsque A = AN.

Il suffit d'écrire :

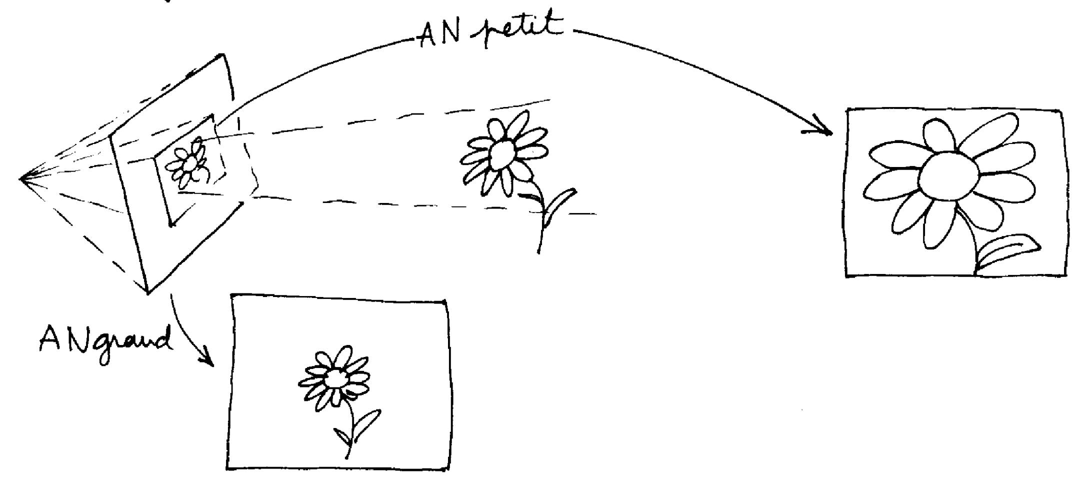
$$X = 140 - A * 139 / AN$$
$$Y = 85 - 140 * S / AN$$

➜ Explication sur les signes. L'Apple II porte, sur l'écran, les Y vers le bas. Et il ne faut pas oublier que si on suit, sur l'écran, un trajet dans le sens des X croissants, ceci correspondra du fait de l'orientation trigonométrique, à des Azimuts décroissants.

L'ouverture angulaire, c'est le **ZOOMING**. En variant AN on aura un effet équivalent au ZOOM :



En accroissant l'ouverture angulaire AN on accroit la portion d'espace qui est perçue, donc on diminue l'importance relative de l'objet sur l'écran, qui apparait donc s'éloigner.

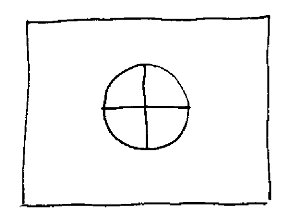


ASTIGMATISME

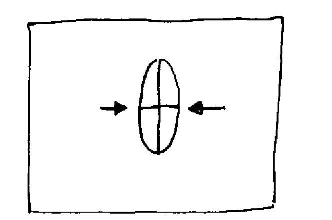
on pourrait tout aussi bien écrire :

$$\begin{cases} X = 140 + A * 139 / AX \\ Y = 85 - S * 140 / AY \end{cases}$$

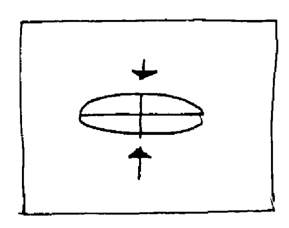
En jouant sur AX et AY on peut introduire une dilatation ou une contraction de l'image selon X ou Y.



AX = AY

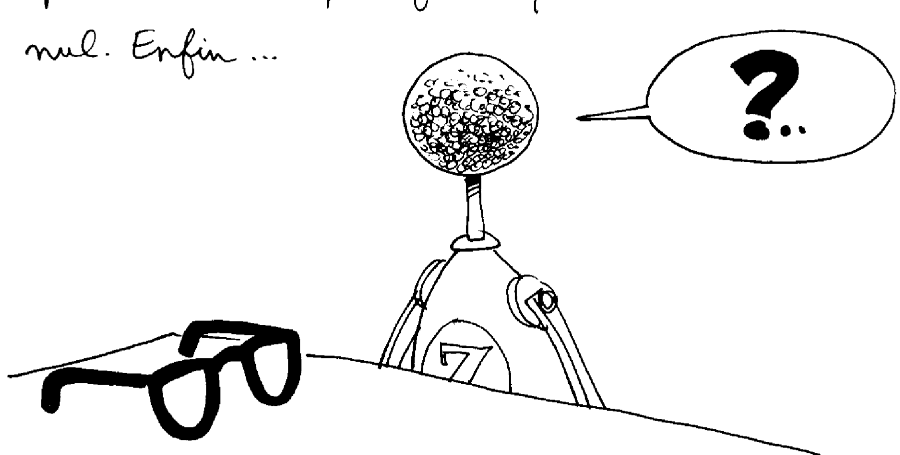


Augmenter AX



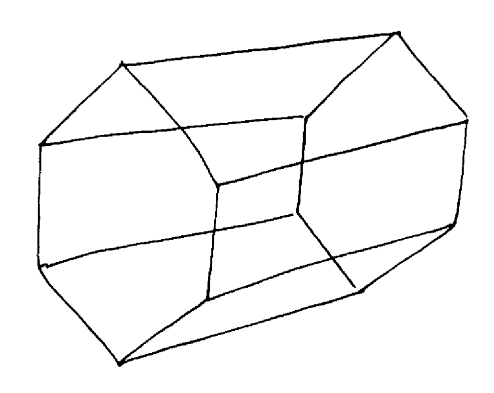
Augmenter AY

Nous nous sommes aperçus que les hommes pouvaient posséder naturellement des altérations de sphéricité de leurs cristallins qui produisaient des effets semblables. Je ne comprend pas à quoi cela pouvait leur servir, d'autant plus qu'en général ils mettaient devant leurs pupilles des verres présentant des variations de courbure inverse, dont l'effet annulait le précédent. Ce qui fait que le résultat était finalement nul. Enfin ...



CODER LES OBJETS

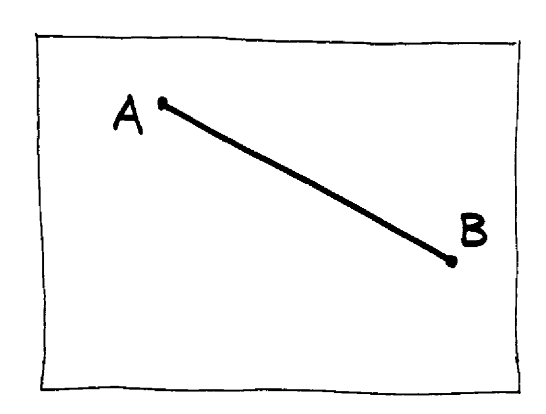
Je me suis vite rendu compte qu'avec une **MATRICE** de quelques 190 par 280 points on ne pouvait pas faire grand' chose. La meilleure solution est de décomposer les objets en **FACETTES** en les représentant comme des structures en "**FIL DE FER**".



Dans ces conditions le geste élémentaire du **DESSIN** consiste à représenter une suite de **SEGMENTS DE DROITE**.

Si **A** est le début du segment et **B** sa fin, et si (**XA**, **YA**) (**XB**, **YB**) sont leurs **COORDONNÉES ÉCRAN** respectives, le tracé sera réalisé suivant :

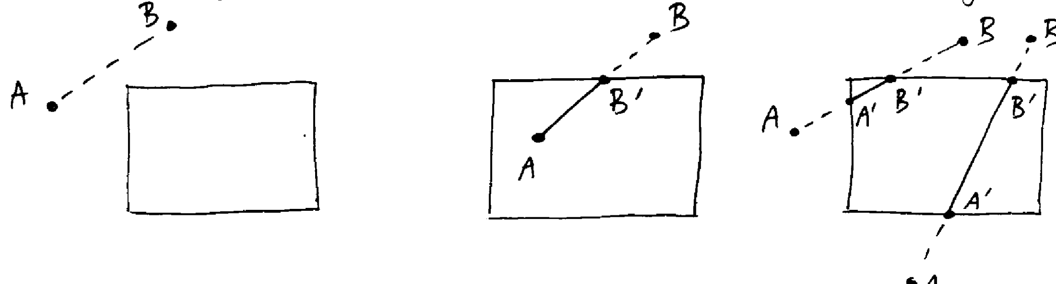
```
HPLOT XA,YA TO XB,YB
```



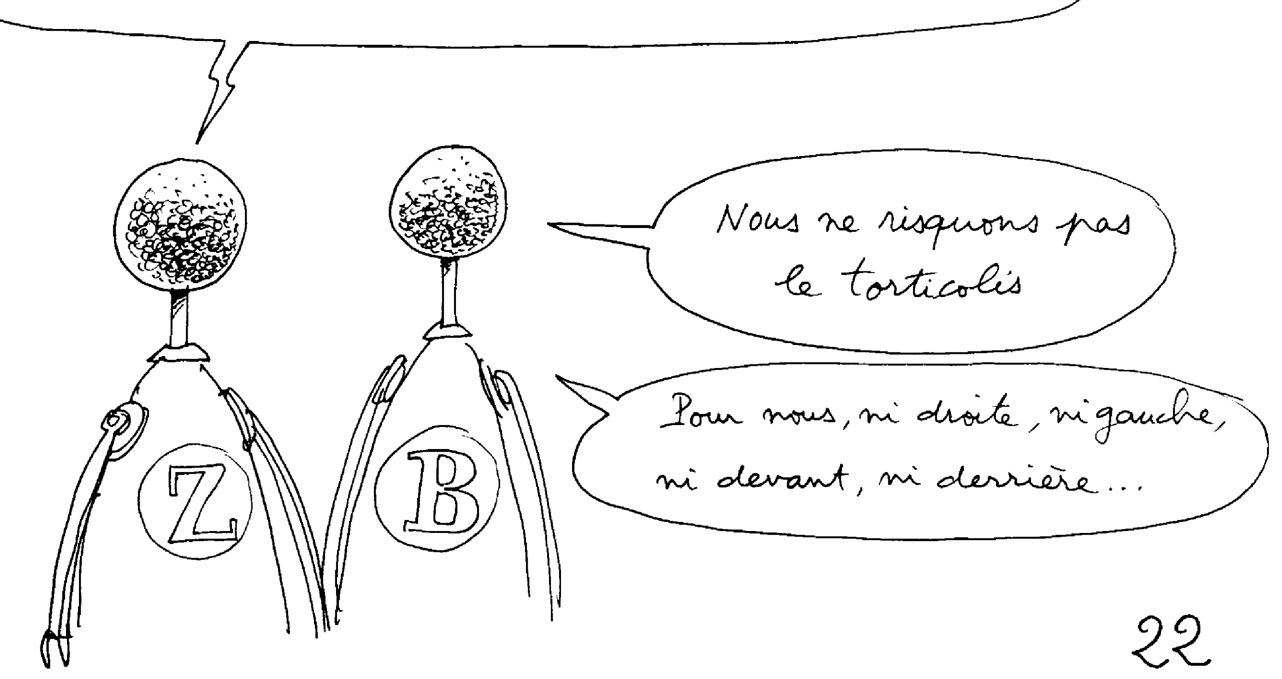
Sur l'APPLE II tout tracé hors écran débouche sur un **MESSAGE D'ERREUR**. Il faudra donc que :

$$\emptyset \leq XA < 28\emptyset$$
$$\emptyset \leq XB < 28\emptyset$$
$$\emptyset \leq YA < 191$$
$$\emptyset \leq YB < 191$$

Cette perception des choses pose un problème de **FENÊTRAGE**. Il faudra que le segment AB ne coupe pas le cadre. Si c'est le cas, il faudra introduire l'intersection segment-cadre :

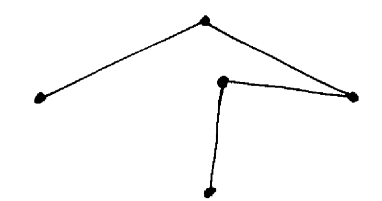


Bah, j'examinerai tout cela plus loin (page 99).



CHAINES

Bon. Il faut maintenant décomposer les objets en contours polygonaux gauches, que j'appelerai chaines :



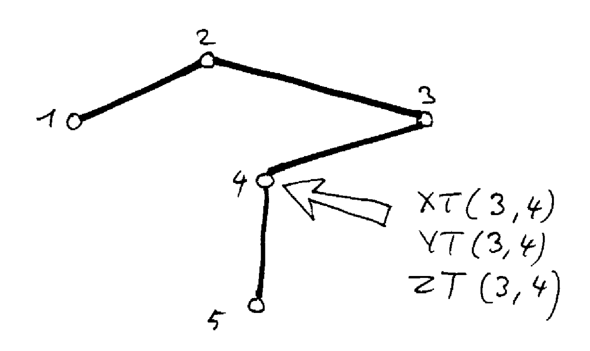
Si un objet a été convenablement décomposé en **CHAINES**, l'opération de **DESSIN** consistera à les représenter les unes après les autres. Un objet sera donc un ensemble de trois **FICHIERS**

XT (I, J)
YT (I, J)
ZT (I, J)

I étant l'indice de la chaine et J l'indice du point sur cette chaine. Par exemple, les coordonnées

XT (3, 4)
YT (3, 4)
ZT (3, 4)

seront celles du quatrieme point de la troisième chaine :



BURP !

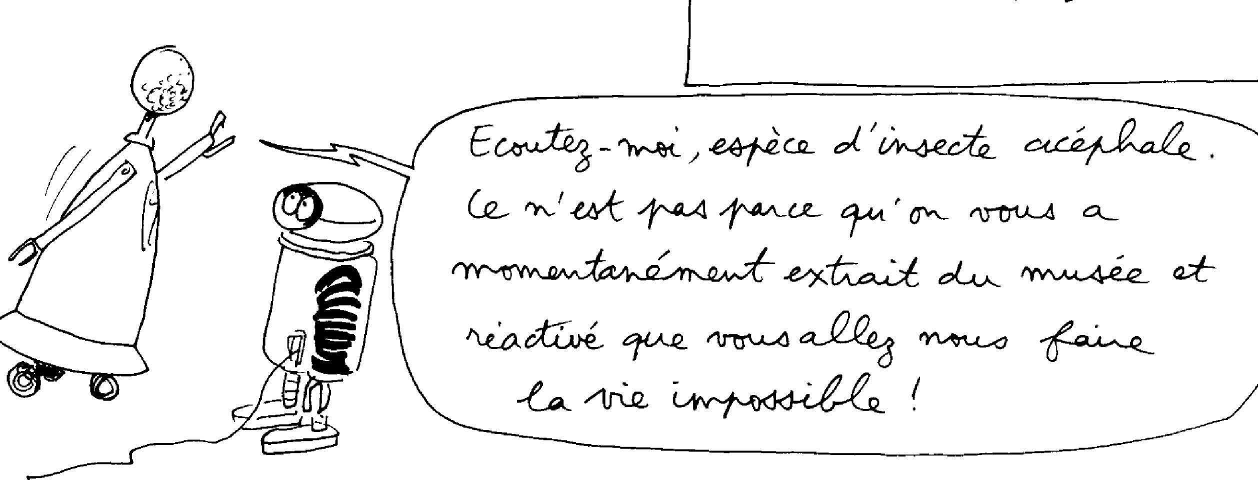
Voyons... qu'est-ce qu'on a comme place-mémoire sur ce bac ? quarante huit K !.. c'est un mauvais rêve... où je vais avec ça ?..

ARTHUR

Tu as vu ? j'ai remonté le petit. Il a l'air de marcher normalement

Bon sang ! tu ne peux pas le tenir tranquille cinq minutes ?

GBONK!



Ah, je ne sais plus où j'en suis avec tout cela. Cela amusait Bernie de réactiver cette relique. Je ne sais même pas si ce genre de modèle est capable de comprendre ce qu'on lui dit... Personnellement, j'en doute.

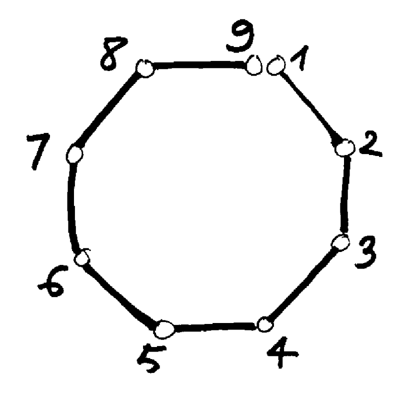
Passons aux réservations de places en mémoire centrale. Voyons... Il y a d'abord ces trois fichiers XT(I,J), YT(I,J), ZT(I,J). Il me faut ensuite un fichier N(I) qui me donnera le nombre de points sur chaque chaine. Et deux fichiers donnant les coordonnées-écran X(I,J), Y(I,J)

Sur l'**APPLE II** on suggère

```
2 DIM XT(3Ø,9),YT(3Ø,9),ZT(3Ø,9),X(3Ø,9),Y(3Ø,9),N(3Ø)
```

qui permet le traitement d'un objet de 30 chaines, comportant au plus neuf points (en fait 31 chaines d'au plus dix points. Mais je n'ai pas compté l'indice zéro).
Cette réservation de place en mémoire, sur un 48K, laissera une place importante pour un programme a multiples fonctions, le **PROGRAMME PANGRAPHE**.

Neuf points permettent la description de cercles selon des **OCTOGONES** (chaine fermée).



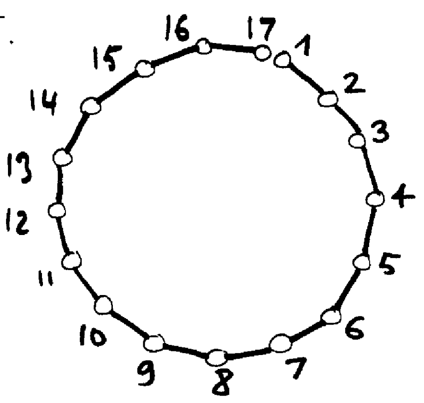
Si le lecteur veut realiser des tracés plus précis, rien n'empêche de mettre plus de points sur les chaines (qui seront moins nombreuses, qui peut le plus peut le moins).

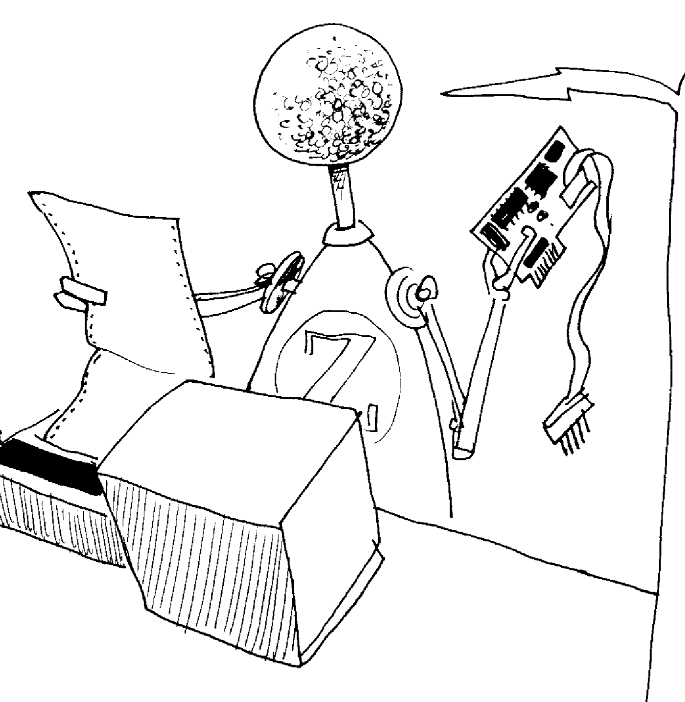
Par exemple :

```
2 DIMXT(15,17),YT(15,17),ZT(15,17),X(15,17),Y(15,17),N(15)
```

Ceci permettant, comme on le verra plus loin, de décrire des cercles à l'aide de 17 points.

Compte tenu des possibilités d'affichage de l'écran, il ne serait guère rentable d'accroître cette définition des cercles.



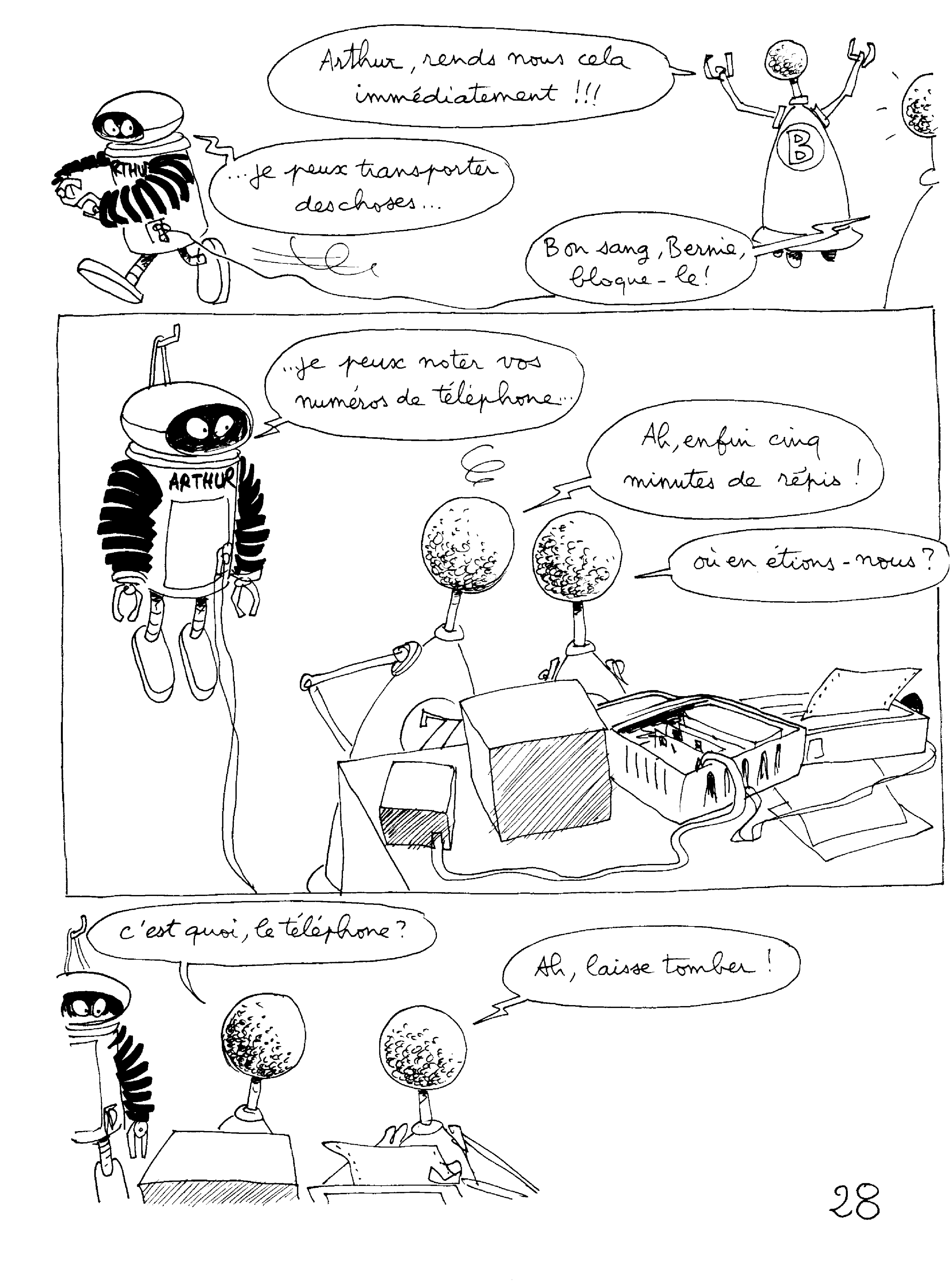




PANGRAPHE

— C'est un logiciel qui a été créé sur un Apple II 48K. Il contient des sections permettant de **CRÉER DES OBJETS**, grâce à une **STRUCTURE D'ACQUISITION DE DONNÉES.**
On peut alors stocker ces objets, qui sont des fichiers de points, des ensembles de **CHAINES**, sur un disque souple, une diskette.




Arthur, rends nous cela immédiatement !!!
...je peux transporter des choses...
Bon sang, Bernie, bloque-le!
ARTHUR
...je peux noter vos numéros de téléphone...
Ah, enfin cinq minutes de répis !
où en étions-nous ?
c'est quoi, le téléphone ?
Ah, laisse tomber !

– **PANGRAPHE OBJET** est piloté par un **PROGRAMME PRINCIPAL**, par l'intermédiaire d'un **MENU**, qui renvoit à tout un ensemble de **SOUS-PROGRAMMES**.

```
0 IF FD = 1 THEN 5
1 L = 0: REM  PANGRAPHE OBJET 7/2/84
2  DIM XT(30,9),YT(30,9),ZT(30,9),X(30,9),Y(30,9),N(30):FD = 1
3 D$ =  CHR$ (13) +  CHR$ (4)
5  TEXT : HOME
6  IF C <  > 0 THEN 310
10  HOME
20  REM

 PROGRAMME MENU


30  VTAB 15: HTAB 15
40  PRINT "VOULEZ-VOUS:"
50  FOR TE = 1 TO 500: NEXT
60  HOME
70  PRINT
80  VTAB 3: HTAB 26: PRINT "PANGRAPHE OBJET"
90  VTAB 1
100  PRINT "1-CREER UN OBJET"
110  PRINT "2-STOCKER UN OBJET"
120  PRINT "3-CHARGER UN OBJET"
130  PRINT "4-COMPLETER UN OBJET"
140  PRINT "5-MODIFIER UN OBJET"
150  PRINT "6-(REPRESENTER UN OBJET)"
160  PRINT "7-(STOCKER UNE IMAGE)"
170  PRINT "8-(CHARGER UNE IMAGE)"
180  PRINT "9-MANIPULER UN OBJET"
190  PRINT "10-DEFINIR UN BLOC D'OBJETS"
200  PRINT "11-CONSULTER UN FICHIER BLOC"
210  PRINT "12-"
220  PRINT "13-"
230  PRINT "14-"
240  PRINT "15-"
250  PRINT "16-"
260  PRINT "17-AFFICHER LE CATALOGUE"
270  PRINT "18-SORTIR DU PROGRAMME"
280  INPUT "VOTRE CHOIX ";C
290  IF C = 17 THEN  PRINT D$"CATALOG": GET A$: GET A$
300  IF C = 18 THEN  END
310  ON C GOSUB 2000,3000,4000,2010,5000,40000,40000,4000
0,17000,28000,29
   000
999  GOTO 10
```

les lignes ø et 6 seront commentées plus loin.

le **MENU** provoque l'affichage sur l'écran d'**OPTIONS**. Certaines options sont indiquées entre parenthèses. Cela aussi sera expliqué plus loin.

L'option 1 (créer un objet) envoit au sous-programme 2øøø.

L'option 2 (stocker un objet) envoit au sous.programme 3øøø

Etc....

La ligne 3 est associée aux **ORDRES DOS**. Tout ordre PRINT suivi de D$, puis d'une instruction DOS sera interprété non comme un ordre d'impression mais comme une instruction du **DISC OPERATING SYSTEM**.

Si l'option 17 est retenue la ligne 290 provoquera l'affichage du CATALOGUE.

CRÉATION D'UN OBJET

Ce sous programme contient un **SOUS-MENU**, qui présente des **SOUS-OPTIONS.**

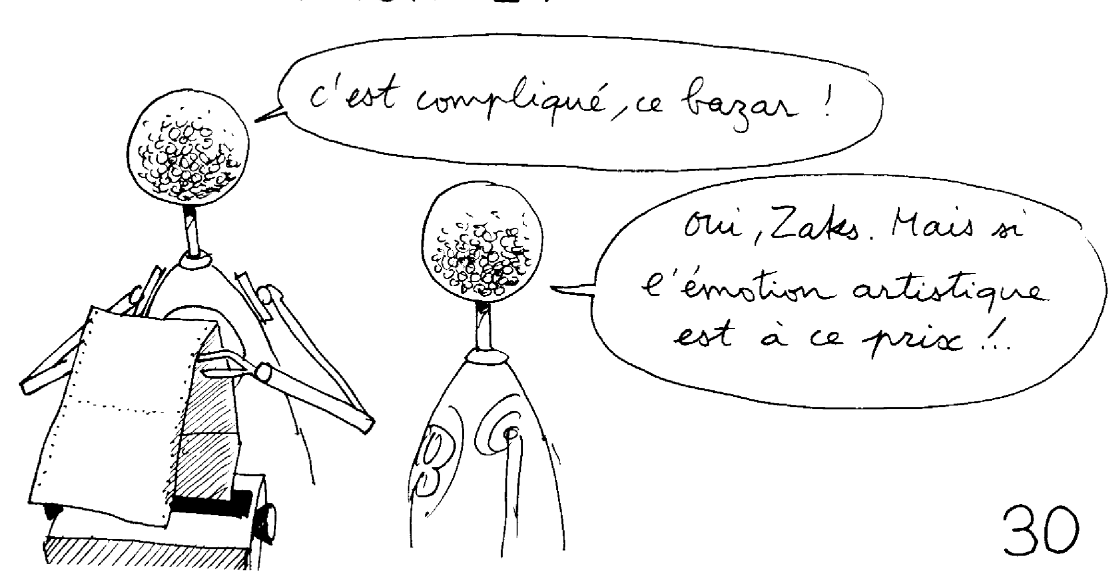
A la ligne 2005 on initialise la variable **L**, qui est l'indice de chaine.

```
2000 REM
CREER UN OBJET

2005 HOME :L = 0
2010 HOME :: PRINT "VOULEZ-VOUS::": PRINT
2020 PRINT "1-CREER DES CHAINES"
2030 PRINT "2-POINTILLER DES SEGMENTS"
2040 PRINT "3-CREER UN CERCLE"
2050 PRINT "4-CHAINER DES CERCLES"
2060 PRINT "5-CHAINER DES COUPLES"
2070 PRINT "6-CREER UN ARC DE CERCLE": PRINT
2100 INPUT "VOTRE CHOIX ";C
2110 ON C GOSUB 10000,11000,12000,13000,14000,1000
2999 RETURN
```

CRÉATION DE CHAINES

– c'est la **SOUS OPTION 1** :



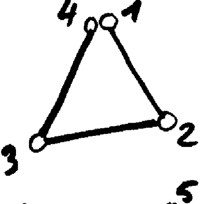
```
10000 REM
 CREATION DE CHAINES

10040 L = L + 1
10050 INPUT "NOMBRE DE POINTS? ";N(L)
10060 J = 0
10070 J = J + 1
10080 PRINT "POINT NO ";J
10090 INPUT "XT=";XT(L,J)
10100 INPUT "YT=";YT(L,J)
10110 INPUT "ZT=";ZT(L,J)
10120 IF J = N(L) THEN 10140
10130 GOTO 10070
10140 INPUT "UNE ERREUR?";R$
10150 IF  LEFT$ (R$,1) = "O" THEN 10050
10160 INPUT "UNE AUTRE CHAINE? ";R$
10170 IF  LEFT$ (R$,1) = "O" THEN 10040
10999 RETURN
```

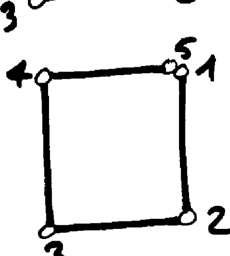
Une petite astuce pour décaler le texte d'un REM (voir page précédente) ;
Lorsqu'on entre le programme dans l'Apple II, immédiatement après le REM, frapper deux fois J^c (control J), puis le texte, puis à nouveau deux J^c, et enfin presser sur la Touche Return.
Ce faisant on a inscrit des **CARACTÈRES DE CONTRÔLE J^c** qui entrainent au **LISTING** des **SAUTS DE LIGNE**, mais sans apparaitre explicitement. Cette astuce permet une relecture facilitée des programmes.

– Ça c'est l'**ENTRÉE DE DONNÉES BESTIALE**. On commence, à la ligne 10040 par incrémenter l'indice de chaine L. Puis l'ordinateur demande le nombre de points de la chaine.

– Si on a un contour polygonal fermé il ne faut pas oublier de le nombre de points de la chaine est égal au nombre de sommets du polygone, augmenté d'une unité.



triangle ⟹ 4 points



quadrangle, carré ⟹ 5 points.

On opère alors la saisie des XT, YT, ZT successifs. Ce qui suppose que l'on aura fait le travail préalable consistant à coter complètement les points de l'objet sur une **ÉPURE**.

Le test 1Ø12Ø permet une **SORTIE DE LA BOUCLE DE SAISIE.** Si on a fait une erreur (cela arrive souvent), le questionnement 1Ø14Ø permet de recommencer la séquence d'acquisition de cette chaine.

On peut, en restant dans ce sous programme 1ØØØØ, entrer plusieurs chaines polygonales, les unes après les autres.

Si on ne désire pas entrer pour le moment d'autre chaine, le RETURN renverra au menu, ou plutôt au sous menu "CRÉATION D'OBJET", lequel renverra à son tour au menu. Si on désire continuer cette création d'objet, on s'orientera vers l'option :

4 - COMPLETER UN OBJET

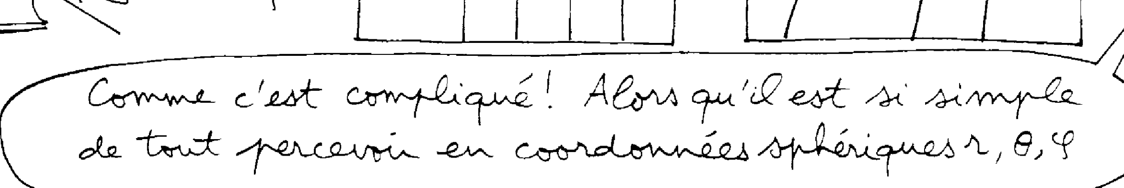
qui, comme le montre la ligne 21Ø, envoit à la ligne 2Ø1Ø. Ce sous programme 2Ø1Ø ... 2999 "shunte" la ligne 2ØØ5 c'est à dire la remise à zéro de l'indice de chaine L. Avec les sous options proposées on pourra créer la chaine L+1, etc...

On verra de nouveau s'afficher le sous-menu :

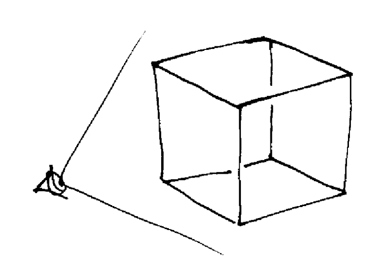
```
VOULEZ-VOUS :

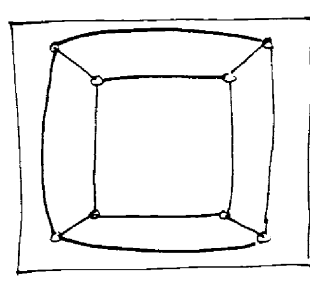
1-CREER DES CHAINES
2-POINTILLER DES SEGMENTS
3-CREER UN CERCLE
4-CHAINER DES CERCLES
5-CHAINER DES COUPLES
6-CRÉER UN ARC DE CERCLE
  VOTRE CHOIX ?
```

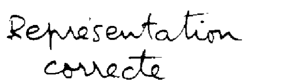


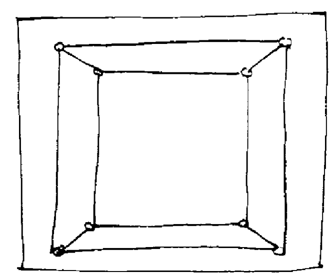


– Si on veut représenter un cube en vision rapprochée, en se contentant des images de ses huit sommets, on fera une erreur :



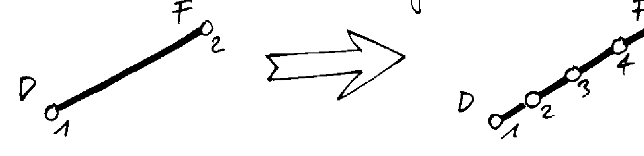






Représentation erronée.

– Pour éviter cette erreur il faut si besoin est **POINTILLER** les segments très proches de l'œil. Pour ce faire (sous-programme 11000) on indiquera les coordonnées du début du segment (XD, YD, ZD), celles de sa fin (XF, YF, ZF). Le programme transformera alors automatiquement ce segment $\overrightarrow{DF}$ en une chaîne rectiligne de cinq points :



```
11000 REM

POINTILLER UN SEGMENT

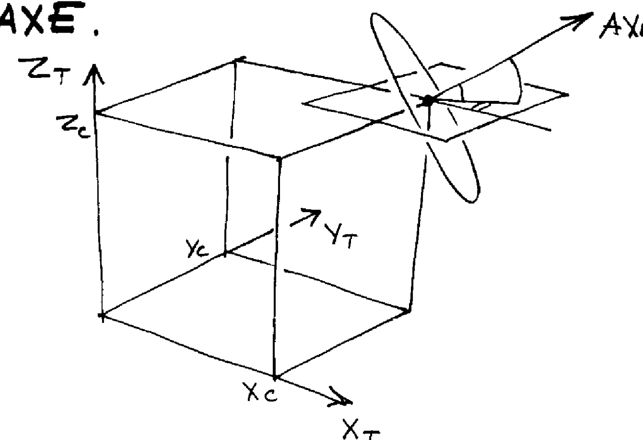
11010 L = L + 1:N(L) = 5
11020  PRINT "DEBUT DU SEGMENT "
11030  INPUT "X=";XD
11040  INPUT "Y=";YD
11050  INPUT "Z=";ZD
11060  PRINT "FIN DU SEGMENT"
11070  INPUT "X=";XF
11080  INPUT "Y=";YF
11090  INPUT "Z=";ZF
11100  FOR W = 1 TO 5
11110 XT(L,W) = XD + (XF - XD) * (W - 1) / 4
11120 YT(L,W) = YD + (YF - YD) * (W - 1) / 4
11130 ZT(L,W) = ZD + (ZF - ZD) * (W - 1) / 4
11140  NEXT W
11999  RETURN
```

Passons maintenant à la :

CRÉATION D'UN CERCLE

C'est la **SOUS-OPTION 3.**

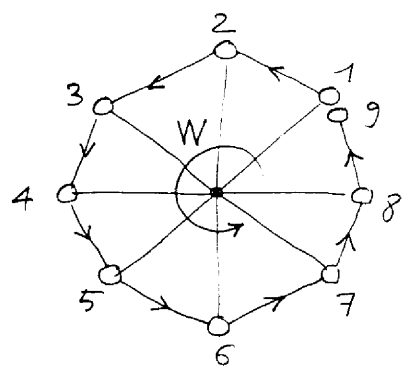
Le cercle sera transformé en **OCTOGONE**. Dans le cas le plus général on le définira par les coordonnées de son centre (**XC, YC, ZC**), son rayon **R** et par l'azimut et le site de l'**AXE**.

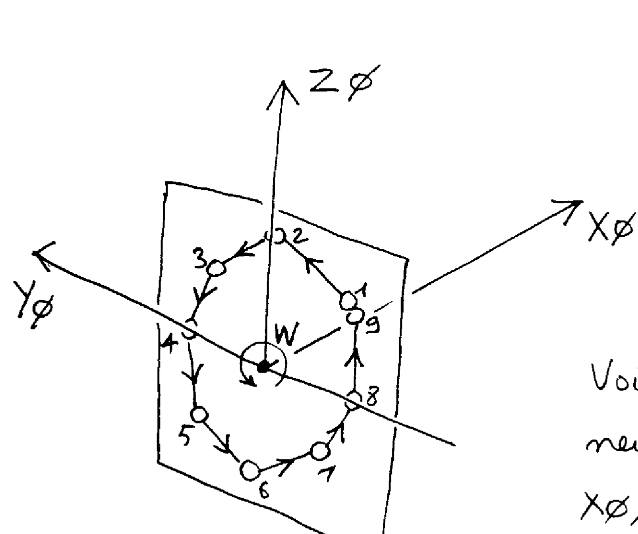


```
12000  REM
CREATION D'UN CERCLE

12010  HOME
12015  PRINT "CENTRE DU CERCLE:": PRINT
12020  PRINT : INPUT "XC=";XC
12030  INPUT "YC=";YC
12040  INPUT "ZC=";ZC
12050  PRINT : INPUT "AZIMUT AXE ";AX
12060 AX = AX * 3.1416 / 180
12065  PRINT : INPUT "ANGLE DE SITE AXE:";SI
12066 SI = SI * 3.1416 / 180
12070  PRINT : INPUT "RAYON ";R
12080 CA =  COS (AX):SA =  SIN (AX)
12085 CS =  COS (SI):SS =  SIN (SI)
12090 L = L + 1
12100 N(L) = 9
12110 J = 0
12120 J = J + 1:W = (J - 1) * .785 / 2
12130 X0 = 0:Y0 =  - R *  COS (W):Z0 = R *  SIN (W)
12132 X1 =  - Z0 * SS + X0 * CS
12133 Y1 = Y0
12134 Z1 = Z0 * CS + X0 * SS
12140 XT(L,J) = X1 * CA - Y1 * SA + XC ] Rotation
12150 YT(L,J) = X1 * SA + Y1 * CA + YC ] Azimutale
12160 ZT(L,J) = Z1 + ZC
12170  IF J = 9 THEN 12190
12180  GOTO 12120
12190 L = L + 1
12200 N(L) = 9
12210 J = 0
12220 J = J + 1:W = (J - 1) * .785 / 2
12230 X0 = 0:Y0 =  - R *  COS (W):Z0 =  - R *  SIN (W)
12232 X1 =  - Z0 * SS + X0 * CS
12233 Y1 = Y0
12234 Z1 = Z0 * CS + X0 * SS
12240 XT(L,J) = X1 * CA - Y1 * SA + XC
12250 YT(L,J) = X1 * SA + Y1 * CA + YC
12260 ZT(L,J) = Z1 + ZC
12270  IF J = 9 THEN 12999
12280  GOTO 12220
12999  RETURN
```

Il y a donc neuf points, numérotés de J = 1 à J = 9.
0,78539 est l'équivalent, en radians, d'un angle de 45° c'est à dire $\frac{45 \times \pi}{180} = 0,78539$.





Voici le positionnement de ces neuf points dans le système d'Axe XØ, XØ, ZØ.

On opère ensuite une rotation suivant OZ d'un angle AX

$$X0 \longrightarrow X0 \cos(AX) - Y0 \sin(AX)$$
$$Y0 \longrightarrow X0 \sin(AX) + Y0 \sin(AX)$$
$$Z0 \longrightarrow Z0$$

Puis on effectue la translation

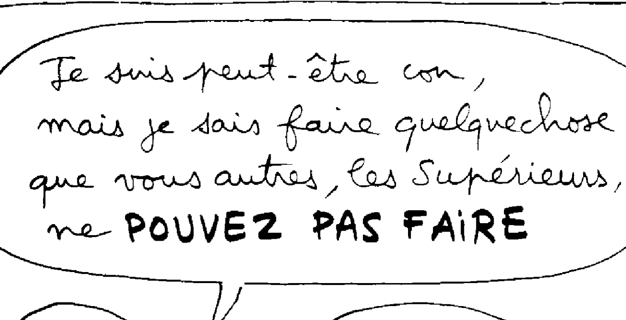
$$\begin{cases} X0 \cos(AX) - Y0 \sin(AX) & \longrightarrow \text{idem} + XC \\ X0 \sin(AX) + Y0 \cos(AX) & \longrightarrow \text{idem} + C \\ Z0 & \longrightarrow \text{idem} + ZC \end{cases}$$

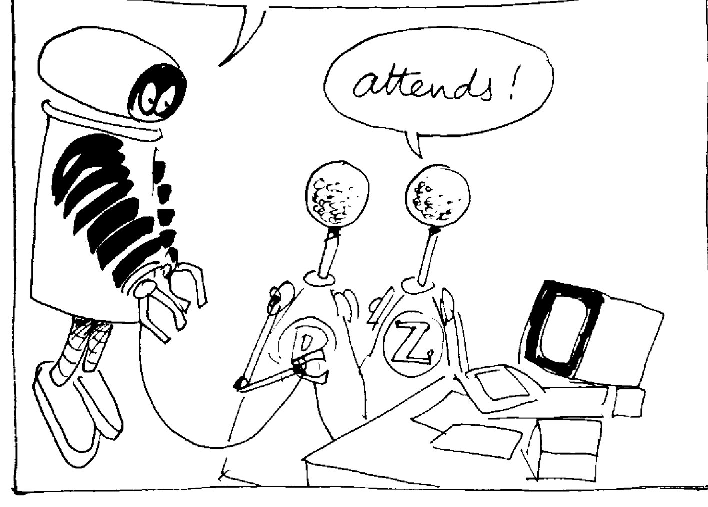
Ce sont ces transformations que traduisent les lignes 12130, 12140, 12150, 12160.

On donnera ultérieurement une autre version de ce sous-programme 12000, plus complète.

OBJETS DE RÉVOLUTION

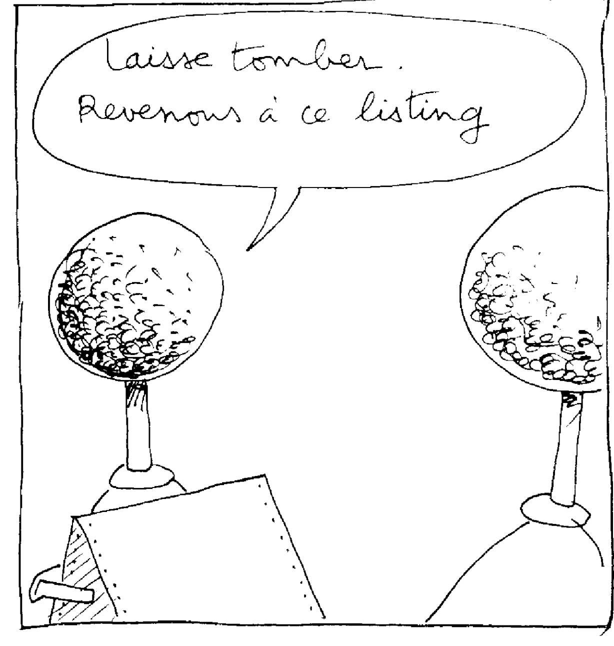












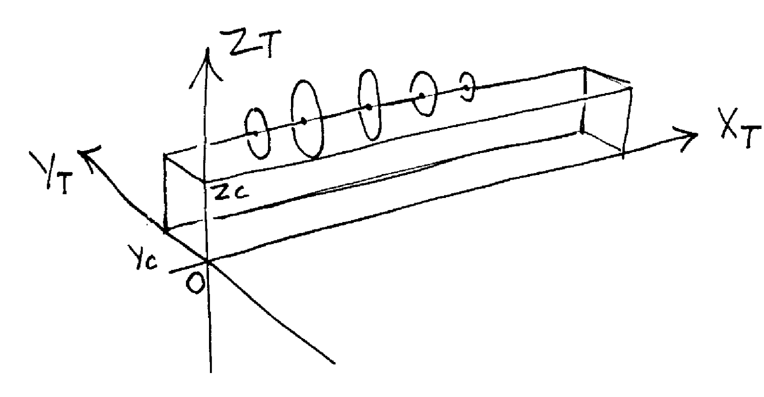
Bon.... Il est intéressant de pouvoir engranger des objets de révolution.

Voici un sous-programme qui permet de créer des octogones coaxiaux, et de chainer les sommets correspondants les uns aux autres. L'axe des octogones étant parallèle à $\overrightarrow{OXT}$.

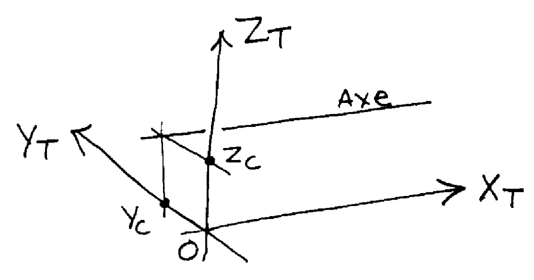
```
13000  REM

CHAINAGE DE CERCLES

13010  HOME : PRINT "CERCLES COAXIAUX // OX"
13012  PRINT : PRINT "AXE DES CERCLES": PRINT
13013  INPUT "YC=";YC
13014  INPUT "ZC=";ZC
13020 K = 0
13030 K = K + 1: PRINT : PRINT "CERCLE NO ";K
13040 L = L + 1:N(L) = 9
13050  INPUT "ABCISSE ";XC
13060  INPUT "RAYON ";R
13070  FOR J = 1 TO 9
13080 W = J * .785
13090 XT(L,J) = XC
13100 YT(L,J) = YC - R *  COS (W):ZT(L,J) = ZC + R *  SIN (W)
13110  NEXT J
13120  INPUT "UN AUTRE?";R$
13130  IF R$ = "OUI" THEN 13030
13140  FOR I = 1 TO 8
13150 L = L + 1:N(L) = K
13160  FOR J = 1 TO K
13170 XT(L,J) = XT(L - K - I + J,I)
13180 YT(L,J) = YT(L - K - I + J,I)
13190 ZT(L,J) = ZT(L - K - I + J,I)
13200  NEXT J
13210  NEXT I
13999  RETURN
```



En 13Ø13 et 13Ø14 on définit la trace de l'axe de l'objet de révolution, parallèle à l'axe $\overrightarrow{OX_T}$, sur le plan $Y_T O Z_T$.



K est l'indice du cercle méridien. L'incrémentation est effectuée à la ligne 13Ø3Ø. Ce nouveau cercle est une nouvelle chaine, d'où incrémentation de L (indice de chaine) dans la ligne 13Ø4Ø.

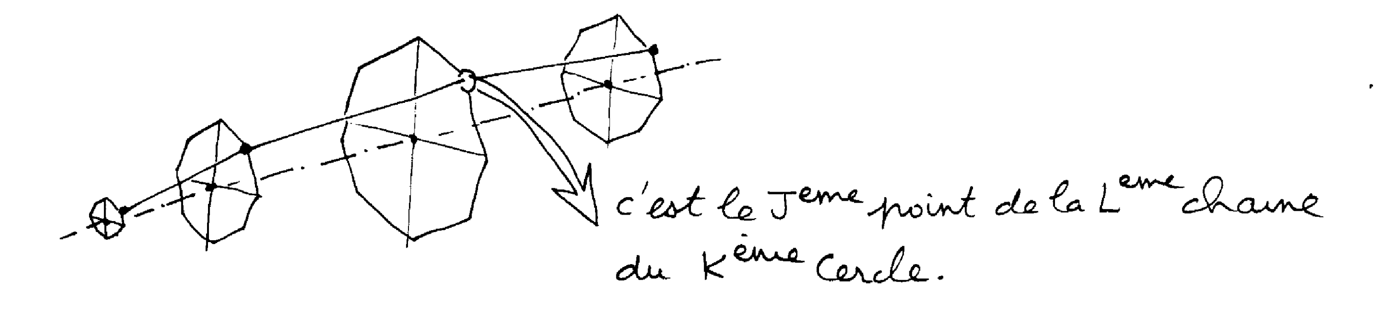
En 13Ø5Ø et 13Ø6Ø on définit l'abcisse X_C et le rayon R du cercle considéré.

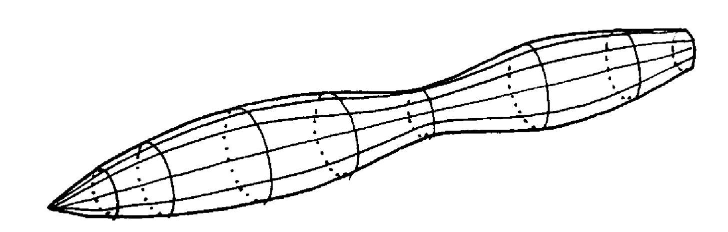
De la ligne 13Ø7Ø à la ligne 1311Ø on "crée" le cercle sous la forme d'une chaine octogonale.

Quand tous ces cercles, ces octogones, sont crées, on va les "chainer".

Ceci débute par un FOR I = 1 TO 8

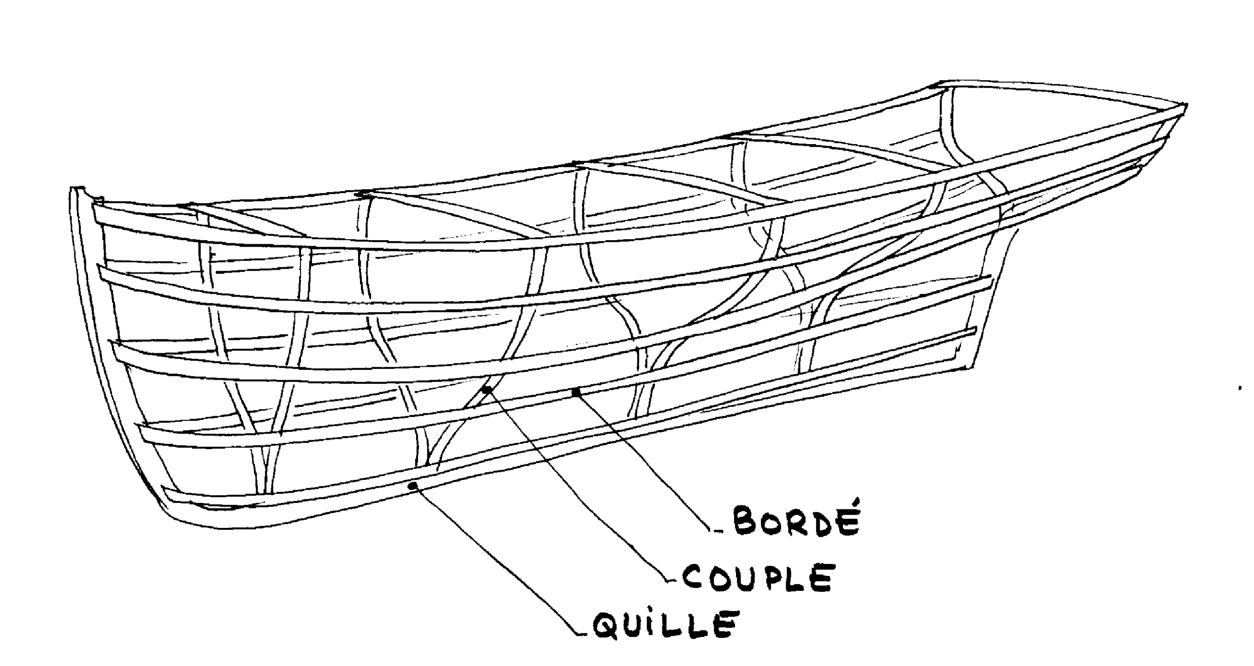
Il y a en effet huit sommets à lier, de proche en proche, sur K cercles.





Il y a beaucoup de façons de créer des objets. Aucun programme ne pourrait les contenir toutes. Chaque discipline, chaque secteur (chimie, architecture, construction navale) demande des procédures d'acquisition spécialement adaptées. Le sous programme ci-après est un exemple de procédure spécialisée, attachée à une variante de **PANGRAPHE** nommée **NAVIGRAPHE**.

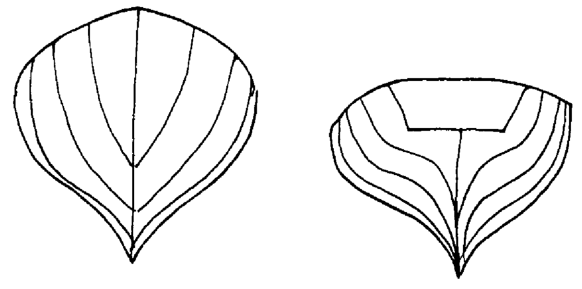
Les coques de navires sont constituées de **COUPLES** et de **BORDÉS** :



La **QUILLE** sera une simple chaine plane.

L'ensemble **COUPLES - BORDÉS** suggère un chaînage analogue au précédent.

Le **PLAN DE FORME** d'un navire est basé sur le dessin d'un certain nombre de couples :



On représentera, bien sûr, ces couples à l'aide de contours polygonaux.

```
14000 REM
CHAINER COUPLES

14010 PRINT "COUPLES AXES SUR OX"
14020 K = 0:L = 0
14030 K = K + 1:L = L + 1:N(L) = 5
14050 PRINT "COUPLE NUMERO ";K
14070 PRINT "DONNEZ 5 PTS :"
```

SAISIE DES POINTS CONSTITUANT LE COUPLE K

```
14080 FOR J = 1 TO 5
14090 PRINT "POINT NUMERO ";J
14100 INPUT "X=";XT(L,J)
14105 INPUT "Y=";YT(L,J)
14110 INPUT "Z=";ZT(L,J)
14120 NEXT J
14130 INPUT "UNE ERREUR? ";R$
14140 IF R$ = "OUI" THEN 14050
14230 INPUT "UN AUTRE COUPLE?";R$
14240 IF R$ = "OUI" THEN 14030
```

FABRICATION DES BORDÉES

```
14260 FOR I = 1 TO 5
14270 L = L + 1:N(L) = K
14280 FOR J = 1 TO K
14290 XT(L,J) = XT(J,I)
14300 YT(L,J) = YT(J,I)
14310 ZT(L,J) = ZT(J,I)
14320 NEXT J: NEXT I
14331 FOR I = 1 TO K: FOR J = 1 TO 5
14332 PRINT XT(I,J),YT(I,J),ZT(I,J)
14333 NEXT J: NEXT I
14334 GET A$
14340 GOSUB 3000
```

FABRICATION DE L'OBJET SYMÉTRIQUE PAR RAPPORT AU PLAN X_TOZ_T

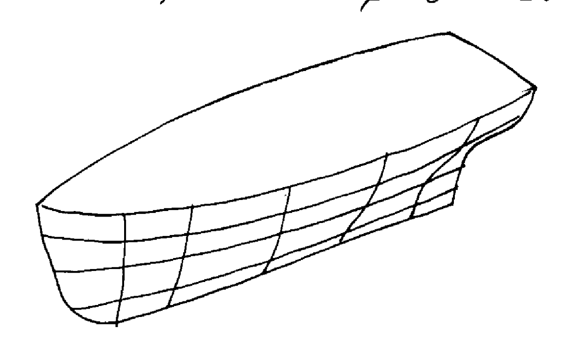
```
14350 FOR I = 1 TO K + 5
14360 FOR J = 1 TO N(I)
14370 YT(I,J) = - YT(I,J)
14380 NEXT J: NEXT I
14400 GOSUB 3000
14999 RETURN
```

couple N° K

On peut tout d'abord indiquer le nombre NB de points constituant le **DEMI-COUPLE**.

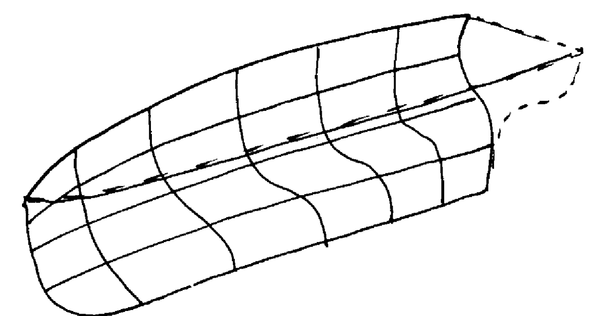
De 14060 à 14100 on saisit les NB sommets de cette ligne brisée qui n'est pas nécessairement plane.

De 14260 à 14320 on "chaine" ces demi couples :



en fabriquant les **NB BORDÉS**. Un GOSUB 3000 (voir plus loin) permet de STOCKER cet objet qui constitue une **DEMI-COQUE** Tribord ou babord.

De 14350 en 14380 on crée la demi coque symétrique par rapport au plan X_TOZ_T (en changeant YT(I,J) en −YT(I,J))



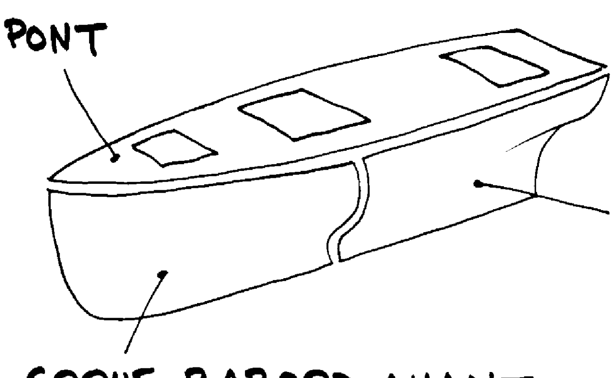
Et on stocke ce nouvel objet.

On verra par la suite que le programme **PANGRAPHE** permet de représenter un objet

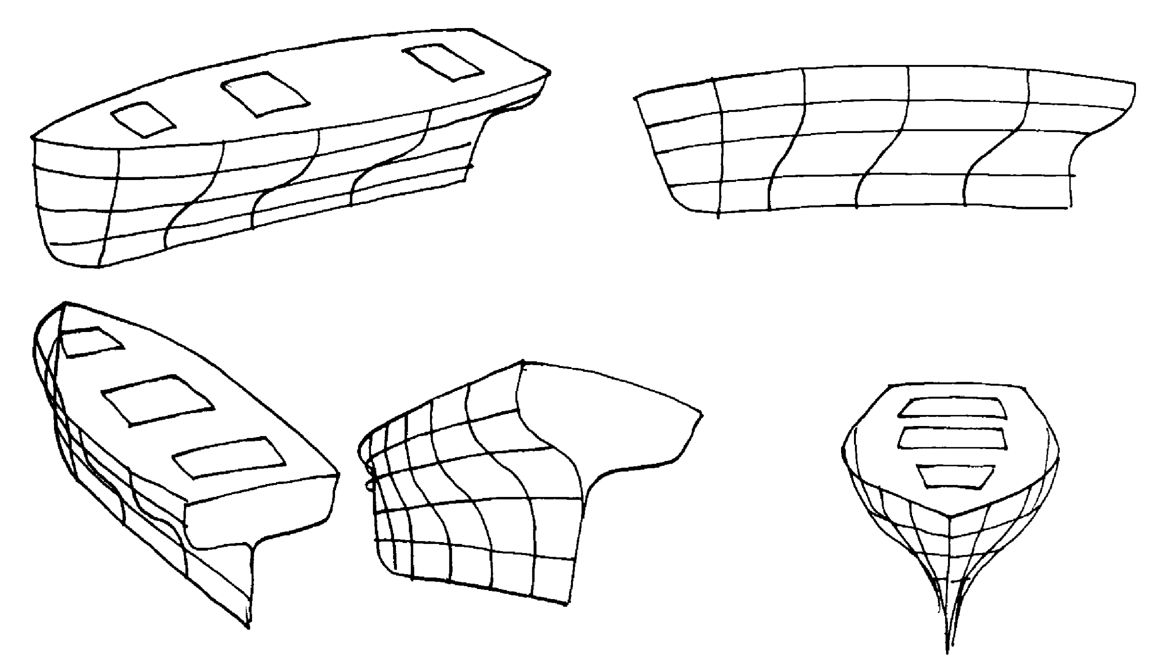
MORCEAU PAR MORCEAU.

Il est donc tout à fait indiqué de décomposer un navire en éléments tels que =

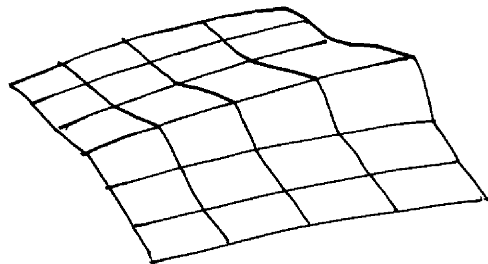
- PONT
- QUILLE
- COQUE BABORD AVANT
- COQUE BABORD ARRIÈRE
- COQUE TRIBORD AVANT
- COQUE TRIBORD ARRIERE



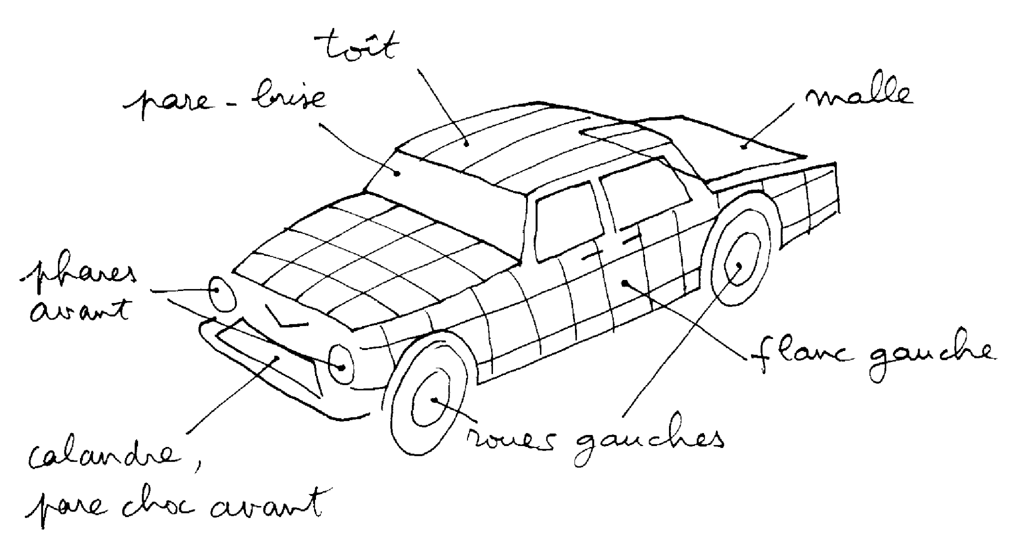
Ce découpage permet, lors de l'opération de dessin en perspective, d'éliminer quelques parties cachées, selon les points de vue :



Si on supprime la section créant le symétrique par rapport au plan $X_T O Z_T$ il est évident que l'on peut créer des éléments de surface maillés, de forme quelconque.



Il est donc possible, en partant d'un plan soigneusement coté, et en effectuant un découpage adéquat - de représenter un objet aussi complexe qu'une automobile avec un micro ordinateur :



STOCKER UN OBJET

Chaque objet, ou élément d'un objet, est un fichier. Quand l'opération de **SAISIE** est achevée, il est indiqué de **STOCKER CET OBJET** sur une diskette. Inversement, le sous programme 4000 réalisera le transfert en mémoire centrale de ce fichier-objet. Nous savons, conformément au **DOS** de l'**APPLE II**, que tout ordre PRINT suivi du caractère D$ = CHR$(4) (défini à la ligne 3) sera interprété non comme un ordre d'impression, mais comme un **ORDRE DOS**

Dans le sous programme 3000, ci-après, on commence par stocker le nombre L de chaines, puis le fichier N(L) donnant le nombre de points dans chaque chaine.

Enfin on passera au stockage des fichiers XT(I,J), YT(I,J), ZT(I,J)

```
3000  REM

STOCKER UN OBJET

3010  INPUT "NOM DE L'OBJET";O$
3020 D$ =  CHR$ (13) +  CHR$ (4)
3030  PRINT D$"OPEN";O$
3040  PRINT D$"WRITE";O$
3050  PRINT L
3060  FOR I = 1 TO L
3070  PRINT N(I)
3080  NEXT I
3090  FOR I = 1 TO L
3100  FOR J = 1 TO N(I)
3110  PRINT XT(I,J): PRINT YT(I,J): PRINT ZT(I,J)
3120  NEXT J: NEXT I
3130  PRINT D$"CLOSE";O$
3999  RETURN
4000  REM
```

CHARGER UN OBJET

```
4000  REM

CHARGEMENT OBJET

4010  INPUT "NOM DE L'OBJET ";O$
4020 D$ =  CHR$ (13) +  CHR$ (4)
4030  PRINT D$"OPEN";O$
4040  PRINT D$"READ";O$
4050  INPUT L
4060  FOR I = 1 TO L
4070  INPUT N(I)
4080  NEXT I
4090  FOR I = 1 TO L
4100  FOR J = 1 TO N(I)
4110  INPUT XT(I,J): INPUT YT(I,J): INPUT ZT(I,J)
4120  NEXT J: NEXT I
4130  PRINT D$"CLOSE";O$
4999  RETURN
```

Ce faisant, nous avons décrit les cinq sous-options du sous-menu "CREER UN OBJET".

Voyons comment

MODIFIER UN OBJET

```
5000  REM

MODIFIER UN OBJET

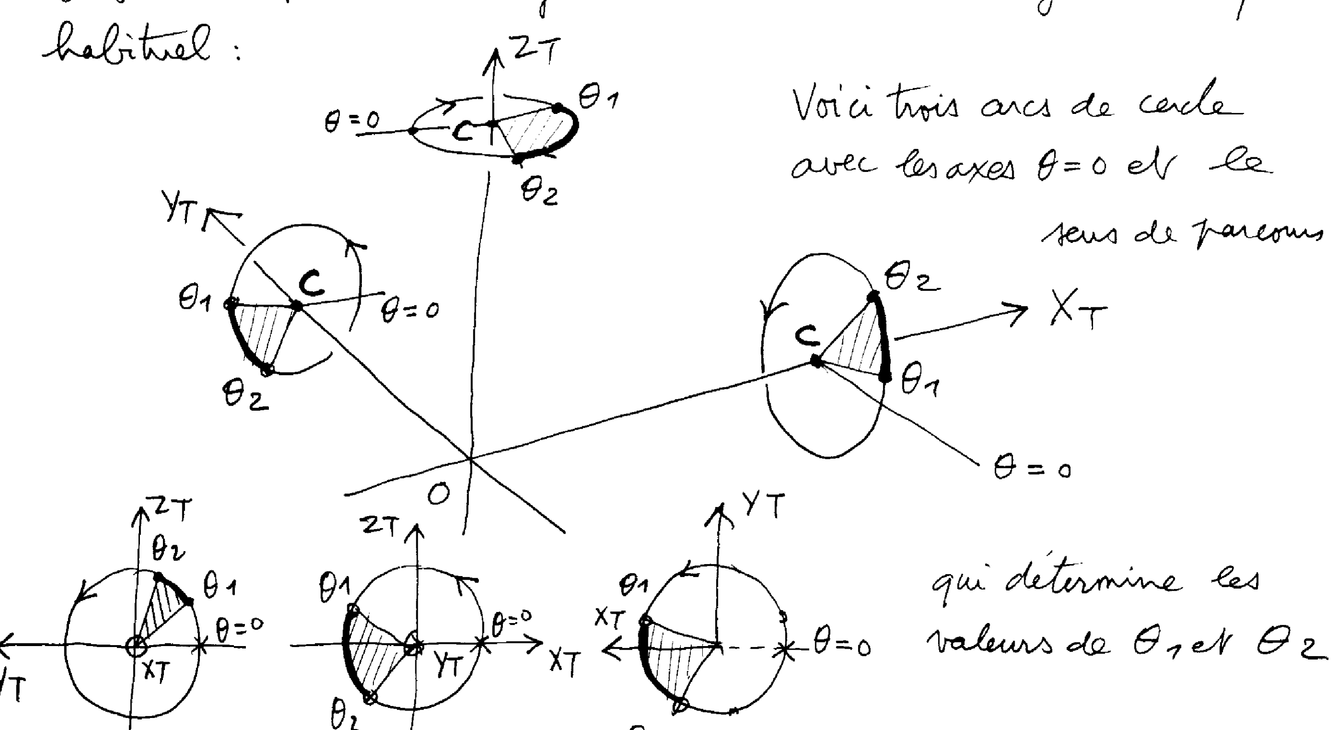
5010  HOME
5020  GOSUB 30000
5030  PRINT : PRINT "OBJET DE ";L;" CHAINES": PRINT
5040  INPUT "DONNEZ L'INDICE DE CHAINE ";I
5050  INPUT "DONNEZ INDICE POINT SUR LA CHAINE ";J
5060  PRINT
5070  INPUT "XT(I,J)=";XT(I,J): PRINT
5080  INPUT "YT(I,J=";YT(I,J): PRINT
5090  INPUT "ZT(I,J)=";ZT(I,J): PRINT
5100  INPUT "UN AUTRE POINT ";R$
5110  IF  LEFT$ (R$,1) = "O" THEN 5040
5120  GOSUB 3020
5998  END
5999  RETURN
```

CRÉER UN ARC DE CERCLE

Dans le sous menu nous avons rajouté une sous option création d'un arc de cercle. Cela sera très utile pour les **CONGÉS**: qui seront alors des quarts de cercle.

On donnera le centre de cet arcle de cercle (XC,YC,ZC) et on se limitera à des arcs d'axes parallèles à $\vec{OX_T}$, $\vec{OY_T}$, $\vec{OZ_T}$.

Le sens de parcours angulaire sera le sens trigonométrique habituel :

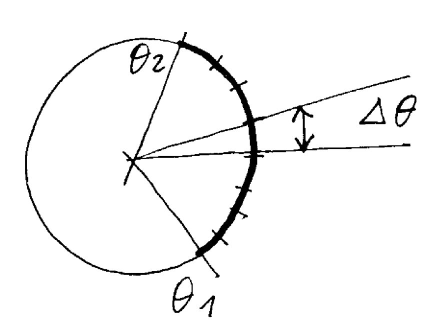


```
1000 REM
CREATION ARC DE CERCLE

1010 HOME :FX = 0:FY = 0:FZ = 0
1015 VTAB 15: HTAB 8
1020 HOME
1030 PRINT "COORDONNEES CENTRE DU CERCLE": PRINT
1040 INPUT "XC=";XC
1050 INPUT "YC=";YC
1060 INPUT "ZC=";ZC: PRINT
1070 PRINT "CERCLE D'AXE PARALLELE A:": PRINT
1080 PRINT "1-OX  2-OY  3-OZ": PRINT
1090 INPUT "VOTRE CHOIX ";C: PRINT
1095 INPUT "RAYON DU CERCLE ";R
1100 IF C = 1 THEN FX = 1
1110 IF C = 2 THEN FY = 1
1120 IF C = 3 THEN FZ = 1
1130 PRINT "SENS DE PARCOURS TRIGO HABITUEL": PRINT

1140 INPUT "DONNEZ THETA 1,DEGRES:";T1
1150 INPUT "DONNEZ THETA 2,DEGRES:";T2: PRINT
1160 T1 = T1 * 3.1416 / 180: REM  CONVERSION DEGRES/RA
DIANS
1170 T2 = T2 * 3.1416 / 180: REM  CONVERSION DEGRES/RA
DIANS
1180 DT = (T2 - T1) / 8
1190 L = L + 1: REM  NOUVELLE CHAINE
1200 N(L) = 9: REM  9 POINTS SUR LA CHAINE
1210  FOR K = 0 TO 8: REM  CREATION ARC DE CERCLE
1220 CT =  COS (T1 + DT * K):ST =  SIN (T1 + DT * K)
1230  REM LIGNES TRIGO DE THETA
1235 J = K + 1
1240 XT(L,J) = XC + (FY + FZ) * CT * R
1250 YT(L,J) = YC + (FX * CT + FZ * ST) * R
1260 ZT(L,J) = ZC + (FX + FY) * ST * R
1270  NEXT K
1275  FOR J = 1 TO 9
1280 XT(L,J) = ( INT (1000 * XT(L,J))) / 1000
1290 YT(L,J) = ( INT (1000 * YT(L,J))) / 1000
1300 ZT(L,J) = ( INT (1000 * ZT(L,J))) / 1000
1400  PRINT XT(L,J),YT(L,J),ZT(L,J)
1410  NEXT J
1420  GET A$
1999  RETURN
```

En 11ØØ, 111Ø, 112Ø on oriente le travail avec des FLAGS ce qui évite une arborescence exccessive des programmes.



l'arc de cercle est tracé selon une chaine à huit côtés et neuf sommets. DTHETA représente l'accroissement de Théta dans le tracé.

la suite 121Ø, 122Ø, 123Ø permet de créer les point XT, YT, ZT dans les trois cas de figure :

- (FX = 1, FY = Ø, FZ = Ø)
- (FX = Ø, FY = 1, FZ = Ø)
- (FX = Ø, FY = Ø, FZ = 1)

En fin de sous programme on affiche les coordonnées calculées pour simple vérification (vous risquez de vous emmêler les pieds dans les orientations et les axes origine).
Et on relance avec un GET A$

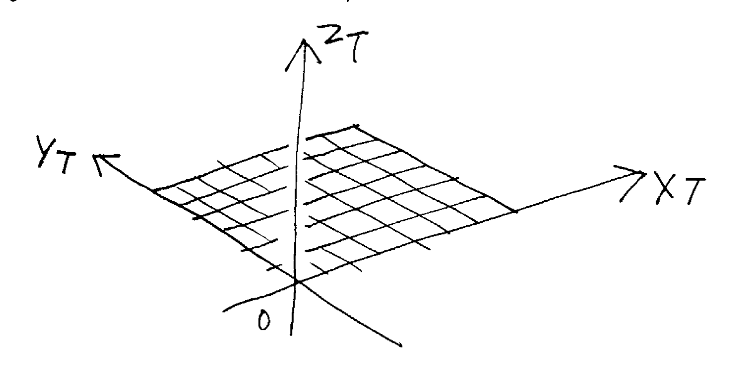
Revenons au **MENU** de la page 67. On y trouve une option
9 - MANIPULER UN OBJET
associée au sous programme 17000

```
17000  REM

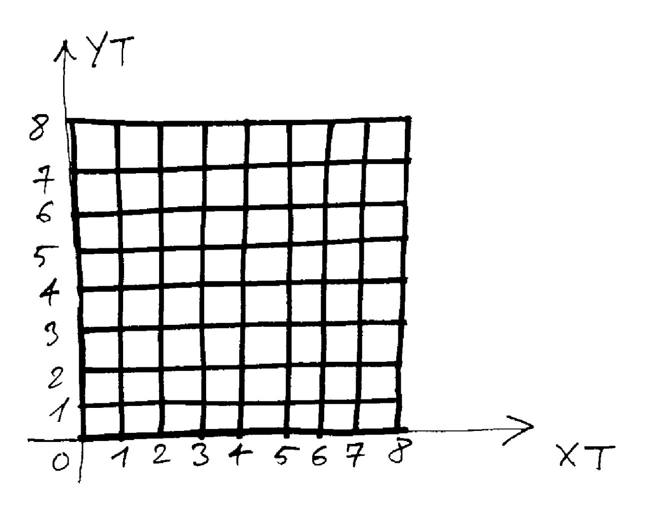
MANIPULER UN OBJET

17010  HOME
17020  PRINT "VOULEZ-VOUS:": PRINT
17030  PRINT "1-OPERER UNE TRANSLATION DE L'OBJET":
PRINT
17040  PRINT "2-OPERER UNE ROTATION": PRINT
17050  PRINT "3-CREER LE SYMETRIQUE % UN PLAN": PRIN
T
17060  PRINT "4-FUSIONNER DEUX OBJETS": PRINT
17070  PRINT "5-HOMOTHETIE": PRINT
17080  PRINT "6-AFFINITE": PRINT
17090  PRINT "7-HOMOTOPIE": PRINT
17100  INPUT "VOTRE CHOIX ";C
17110  ON C GOSUB 19000,20000,18000,21000,22000,23000,240
00
17999  RETURN
```

Dans ce sous menu on a tenté de regrouper un certain nombre d'opérations élementaires, géométriques Pour penser ces manipulations d'objets, il faut avoir un état d'esprit de chaudronnier, ou d'artisan tôlier formeur.
Imaginons un objet qui soit un simple quadrillage :



Voici un court programme servant à créer la grille ci-contre, que nous appelerons GRILLE 1

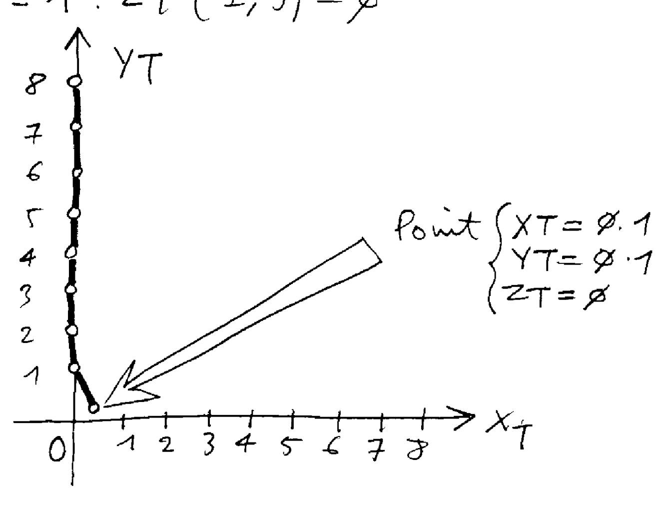



```
1 DIM XT(3Ø,9),YT(3Ø,9),ZT(3Ø,9),N(3Ø)
2 D$=CHR$(4)
```

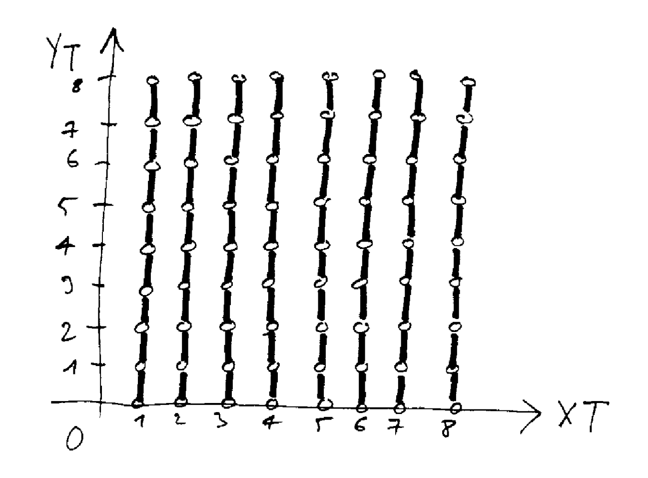
Nous n'allons pas tracer exactement cette grille, passant par le point (Ø,Ø,Ø), mais nous allons très légèrement la déformer pour que ce point (XT = Ø ; YT = Ø ; ZT = Ø) soit remplacé par XT = Ø.1 ; YT = Ø.1 ; ZT = Ø (ce qui nous permettra plus loin avec plus d'aisance de "passer à travers" un objet, en évitant que l'un de ses points ne soit confondu avec le point d'observation. D'où :

```
1Ø XT(1,1) = Ø.1 : YT(1,1) = Ø.1 : Z = Ø
2Ø I = 1 : N(I) = 9          (Neuf points sur la chaine)
3Ø FOR J = 2 TO 9
4Ø XT(I,J) = Ø : YT(I,J) = J - 1 : ZT(I,J) = Ø
5Ø NEXT J
```

création de la chaine indiquée ci-contre, qui va de YT = Ø.1, YT = 1, YT = 2, ⋮ YT = 8



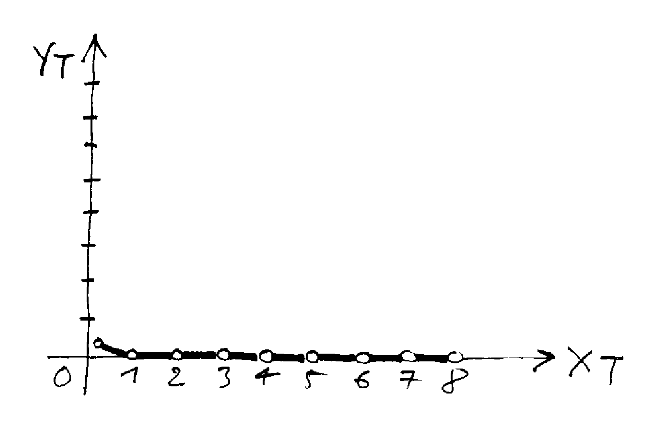
```
60 FOR I=2 TO 9
70 FOR J = 1 TO 9
80 XT(I,J) = I - 1
90 YT (I,J) = J-1
100 ZT(I,J) = 0
110 NEXT J: NEXT I
```



création des huit chaines indiquées ci-dessus.

```
120 XT(10,1) = 0.1  : YT(10,1) = 0.1  : ZT(10,1) = 0
130 N(10) = 9 : I = 9
140 FOR J = 2 TO 9
150 XT(I, J) = J - 1  : YT (I,J) = 0  : ZT (I,J) = 0
160 NEXT J
```

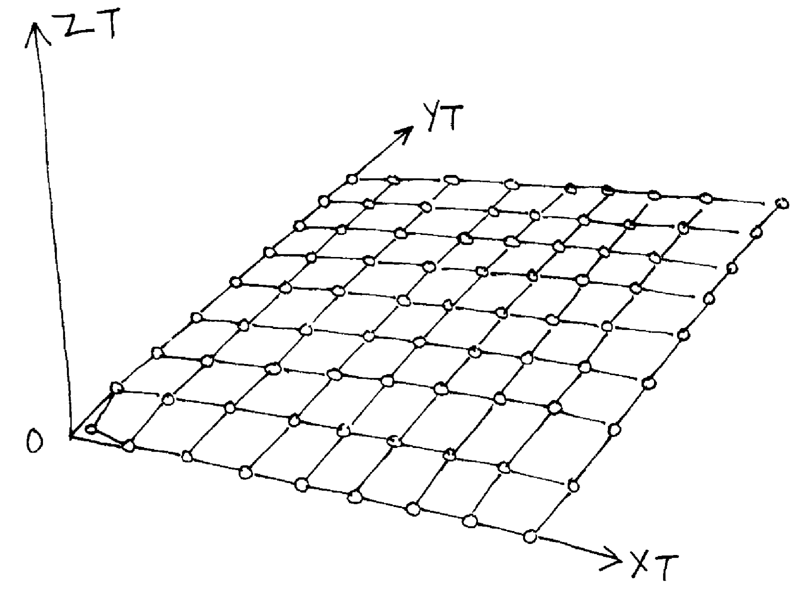
crée :



Enfin :

```
170 FOR I = 11 TO 18   : N(I) = 9
180 FOR J = 1 TO 9
190 XT(I,J) = J-1 : YT(I,J) = I-10  : ZT(I,J) = 0
200 NEXT J : NEXT I
```

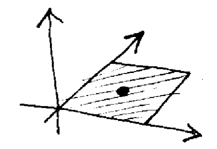
Crée huit autres chaines de neuf points, parallèles à OXT et échelonnées de YT = 1 à YT = 8
L'ensemble donne le maillage ci-après :



Il reste à stocker cet ensemble, que nous appelerons GRILLE1(4-4-∅) sur disque

```
210 PRINTD$"OPEN GRILLE1(4-4-∅)"
220 PRINT D$"WRITE GRILLE1(4-4-∅)"
230 PRINT 18        (nombre de chaines)
240 FOR I=1 TO 18                      (nombre de points par chaine)
250    PRINT 9
260 NEXT I
270 FOR I = 1 TO 18
280 FOR J = 1 TO 9
290 PRINT XT(I,J)
300 PRINT YT(I,J)
310 PRINT ZT (I,J)
320 NEXT J : NEXT I
330 PRINT D$ "CLOSE GRILLE1(4-4-∅)"
340 END
```

A la suite du nom de ce fichier GRILLE1 nous prenons la PRÉCAUTION de faire figurer les coordonnées du centre géométrique de cet objet (point 4,4,0)



Venons-en aux différents sous-programmes de **MANIPULATION D'OBJET.**

TRANSLATION

```
19000 REM

TRANSLATION OBJET

19010  HOME
19020  GOSUB 30000
19040  PRINT "DONNEZ VECTEUR TRANSLATION": PRINT
19050  INPUT "DX=";DX
19060  INPUT "DY=";DY
19070  INPUT "DZ=";DZ
19080  FOR I = 1 TO L
19090  FOR J = 1 TO N(I)
19100 XT(I,J) = XT(I,J) + DX
19110 YT(I,J) = YT(I,J) + DY
19120 ZT(I,J) = ZT(I,J) + DZ
19130  NEXT J: NEXT I
19140  PRINT : PRINT "NOM OBJET TRANSLATE:": PRINT
19150  INPUT "";O$
19160  GOSUB 3030
19999  RETURN
```

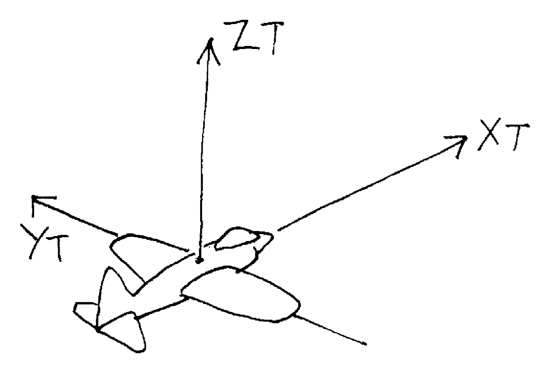
Pas de difficulté particulière.
Passons à la :

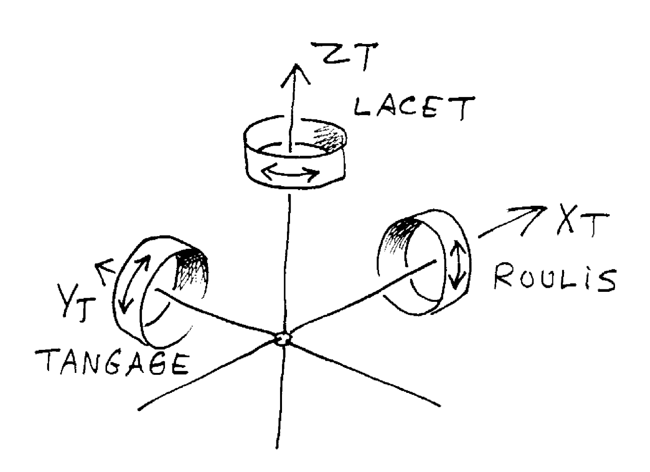
ROTATION

```
20000 REM

ROTATION OBJET

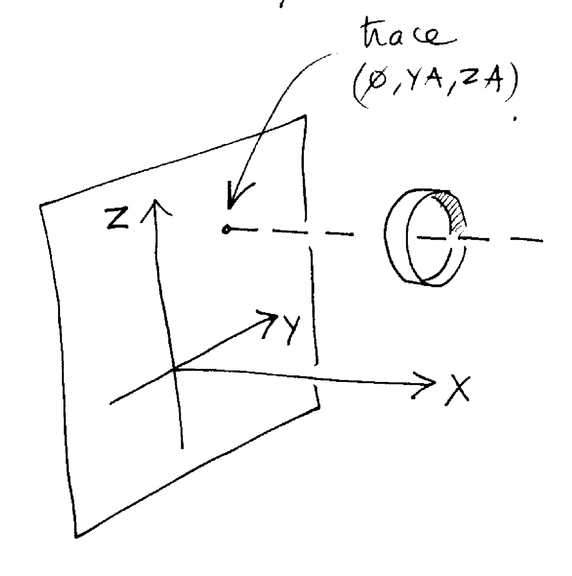
20010  HOME
20020  GOSUB 30000
20025  PRINT
20030  PRINT "ROTATION AXE PARALLELE A:": PRINT
20040  PRINT "1-L'AXE OX : ROULIS": PRINT
20050  PRINT "2-L'AXE OY : TANGAGE": PRINT
20060  PRINT "3-L'AXE OZ : LACET": PRINT
20070  INPUT "VOTRE CHOIX ";C
20080  ON C GOSUB 25000,26000,27000
20999  RETURN
```





Par convention on situera un objet susceptible de mouvement selon l'axe OXT (avion, voiture, etc...)
Par ailleurs une rotation autour d'un axe quelconque est décomposable en une suite de rotations autour de ces axes principaux (qui suffisent largement aux besoins).

ROTATION AUTOUR D'UN AXE PARALLÈLE À L'AXE OX.



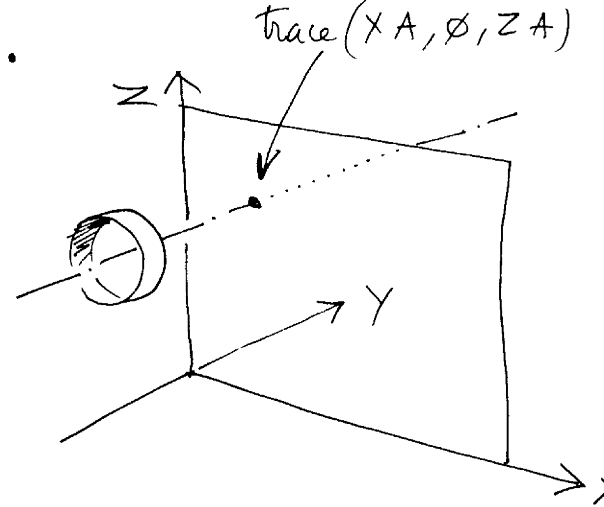

```
25000  REM
ROTATION AXE // OX

25010  HOME
25020  PRINT "TRACE AXE DANS PLAN YOZ": PRINT
25030  INPUT "YA=";YA: PRINT : INPUT "ZA=";ZA: PRINT
25040  INPUT "ANGLE,DEGRES: ";AR
25050  REM  AR ANGLE DE ROULIS
25060 AR = AR * 3.1416 / 180
25070 CR =  COS (AR):SR =  SIN (AR)
25080  FOR I = 1 TO L: FOR J = 1 TO N(I)
25090 X1 = XT(I,J)
25100 Y1 = YT(I,J) - YA              } changement de repère
25110 Z1 = ZT(I,J) - ZA: REM

25120 X2 = X1
25130 Y2 = Y1 * CR - Z1 * SR            } Rotation
25140 Z2 = Y1 * SR + Z1 * CR: REM
25150 XT(I,J) = X2
25160 YT(I,J) = Y2 + YA              } retour au système de
25170 ZT(I,J) = Z2 + ZA: REM           coordonnées initial

25180  NEXT J: NEXT I
25999  RETURN
```

ROTATION AUTOUR D'UN AXE PARALLÈLE A OY.




```
26000  REM

ROTATION AXE // OY

26010  HOME
26020  PRINT "TRACE AXE DANS PLAN XOZ": PRINT
26030  INPUT "XA=";XA: PRINT : INPUT "ZA=";ZA: PRINT
26040  INPUT "ANGLE DEGRES ";A1
26050 A1 = A1 * 3.1416 / 180
26070  REM  A1 ANGLE DE TANGAGE
26075 CT =  COS (A1):ST =  SIN (A1)
26080  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

26100 X1 = XT(I,J) - XA          } changement de coordonnées
26110 Y1 = YT(I,J)
26120 Z1 = ZT(I,J) - ZA: REM

26140 X2 = Z1 * ST + X1 * CT       } Rotation
26150 Y2 = Y1
26160 Z2 = Z1 * CT - X1 * ST: REM
26170 XT(I,J) = X2 + XA          } Retour au système de
26180 YT(I,J) = Y2                 coordonnées XT,YT,ZT
26190 ZT(I,J) = Z2 + ZA: REM

26200  NEXT J: NEXT I
26999  RETURN
```

ROTATION AUTOUR D'UN AXE // À OZ.

```
27000  REM

ROTATION AXE // OZ

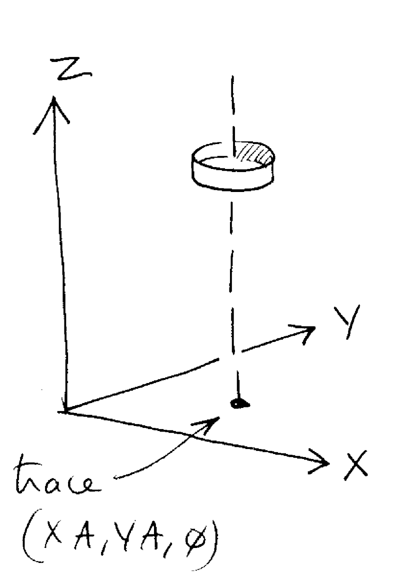
27010  HOME
27020  PRINT "TRACE AXE DANS PLAN XOY": PRINT
27030  INPUT "XA=";XA: PRINT : INPUT "YA=";YA: PRINT
27040  INPUT "ANGLE,DEGRES: ";AL
27050 AL = AL * 3.1516 / 180
27060  REM  AL ANGLE DE LACET
27070 CL =  COS (AL):SL =  SIN (AL)
27080  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

27090 X1 = XT(I,J) - XA   } changement de
27100 Y1 = YT(I,J) - YA     coordonnées
27110 Z1 = ZT(I,J): REM

27120 X2 = X1 * CL - Y1 * SL ] Rotation
27130 Y2 = X1 * SL + Y1 * CL
27140 Z2 = Z1: REM

27150 XT(I,J) = X2 + XA  ] Retour ancien système
27160 YT(I,J) = Y2 + YA     xT, yT, zT
27170 ZT(I,J) = Z2: REM

27180  NEXT J: NEXT I
27999  RETURN
```



SYMÉTRIE PAR RAPPORT À UN PLAN

Nous allons envisager ici trois cas, traduisant les symétries par rapport aux trois plan fondamentaux associés au trièdre de référence.

Pour ne pas multiplier les sous programmes, nous utiliserons ici un système de flags.

```
18000  REM
SYMETRIQUE % PLAN

18001  HOME
18005 FX = 0:FY = 0:FZ = 0
18006  GOSUB 30000
18007  PRINT : PRINT "SYMETRIQUE PAR RAPPORT AU PLAN
": PRINT : PRINT "1-
   Z=0 PLAN XOY": PRINT : PRINT "2- Y=0 PLAN XOZ": PRINT
 : PRINT "3- X=
   0 PLAN YOZ": PRINT : INPUT "VOTRE CHOIX ";C
18008  IF C = 1 THEN FZ = 1
18009  IF C = 2 THEN FY = 1
18010  IF C = 3 THEN FX = 1
18011  FOR I = 1 TO L: FOR J = 1 TO N(I)
18020  IF FX = 1 THEN XT(I,J) =  - XT(I,J)
18022  IF FY = 1 THEN YT(I,J) =  - YT(I,J)
18024  IF FZ = 1 THEN ZT(I,J) =  - ZT(I,J)
18030  NEXT J: NEXT I
18035  PRINT
18040  INPUT "NOM NOUVEL OBJET ";O$
18050  GOSUB 3030
18809  IF C = 2 THEN FY = 1
18999  RETURN
```

Tout ceci est assez simple.

FUSIONNER OBJETS

c'est une opération très simple à opérer. Simple jeu sur les indices. Mais attention ! les fichiers objets de **PANGRAPHE** sont limités à **30 CHAINES.** Soient deux objets. Le premier est constitué de L_1 chaines et le second de L_2 chaines. La fusion des deux fichiers ne pourra s'opérer que si $L_1 + L_2 \leqslant 30$. Bien sûr...

```
21000  REM
FUSION OBJETS

21010  HOME
21020  PRINT "FIN DE FUSION,TAPEZ LE MOT FIN": PRINT
21030  PRINT "PREMIER OBJET A FUSIONNER": PRINT
21035  GOSUB 30000
21040  INPUT "NOM OBJET A FUSIONNER:";O$
21042  IF O$ = "FIN" THEN 21500
21050  PRINT D$"OPEN";O$
21060  PRINT D$"READ";O$
21062  REM
OPERATION DE FUSION

21065  REM  LL=NOMBRE DE CHAINES  DU NOUVEL OBJET
21070  INPUT LL
21080  FOR I = L + 1 TO L + LL
21090  INPUT N(I)
21100  NEXT I
21110  FOR I = L + 1 TO L + LL
21120  FOR J = 1 TO N(I)
21130  INPUT XT(I,J): INPUT YT(I,J): INPUT ZT(I,J)
21140  NEXT J: NEXT I
21150  PRINT D$"CLOSE";O$
21155 L = L + LL
21160  GOTO 21040
21500  REM  FIN DE FUSION
21510  HOME
21520  PRINT : INPUT "NOM DE L'OBJET GLOBAL ";O$
21530  GOSUB 3030
21999  RETURN
```

boucle de saisie des objets à fusionner

On a une boucle de saisie. La sortie de boucle étant réalisée lorsqu'on tente d'appeler l'objet "FIN". L'opération de fusion consiste à aller chercher le fichier des chaines constituant le nouvel objet, et à l'inscrire au bout de l'objet déja présent en mémoire

HOMOTHÉTIE

Voici une transformation géométrique classique qui va permettre de "gonfler" ou de "dégonfler" un objet.

Il faut d'abord définir un **CENTRE D'HOMOTHÉTIE C** qui aura pour coordonnées XC, YC, ZC. A tout point M (XT, YT, ZT) on pourra associer un vecteur $\overrightarrow{CM}$ $\begin{cases} XT - XC = X1 \\ YT - YC = Y1 \\ ZT - ZC = Z1 \end{cases}$

On se donnera ensuite le **COEFFICIENT D'HOMOTHETIE HO**.

Et, au vecteur $\overrightarrow{CM}$ on substituera le vecteur

$$\overrightarrow{CM2} \begin{cases} X2 = X1 * HO \\ Y2 = Y1 * HO \\ Z2 = Z1 * HO \end{cases}$$

Si HO = 1 l'objet est inchangé. Mais, si HO = −1 on réalise une **SYMÉTRIE %. C** (par rapport au Centre C).

L'écriture du sous programme est simple affaire de routine.

```
22000  REM
HOMOTHETIE

22010  HOME
22020  GOSUB 30000
22040  PRINT "COORDONNEES CENTRE HOMOTHETIE": PRIN
T
22050  INPUT "XC=";XC
22060  INPUT "YC=";YC
22070  INPUT "ZC=";ZC
22080  PRINT : INPUT "COEFFICIENT HOMOTHETIE ";HO: PR
INT
22090  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

22100 X1 = XT(I,J) - XC
22110 Y1 = YT(I,J) - YC
22120 Z1 = ZT(I,J) - ZC: REM

22130 X2 = X1 * HO
22135 Y2 = Y1 * HO
22140 Z2 = Z1 * HO: REM

22145 XT(I,J) = X2 + XC
22150 YT(I,J) = Y2 + YC
22160 ZT(I,J) = Z2 + ZC: REM

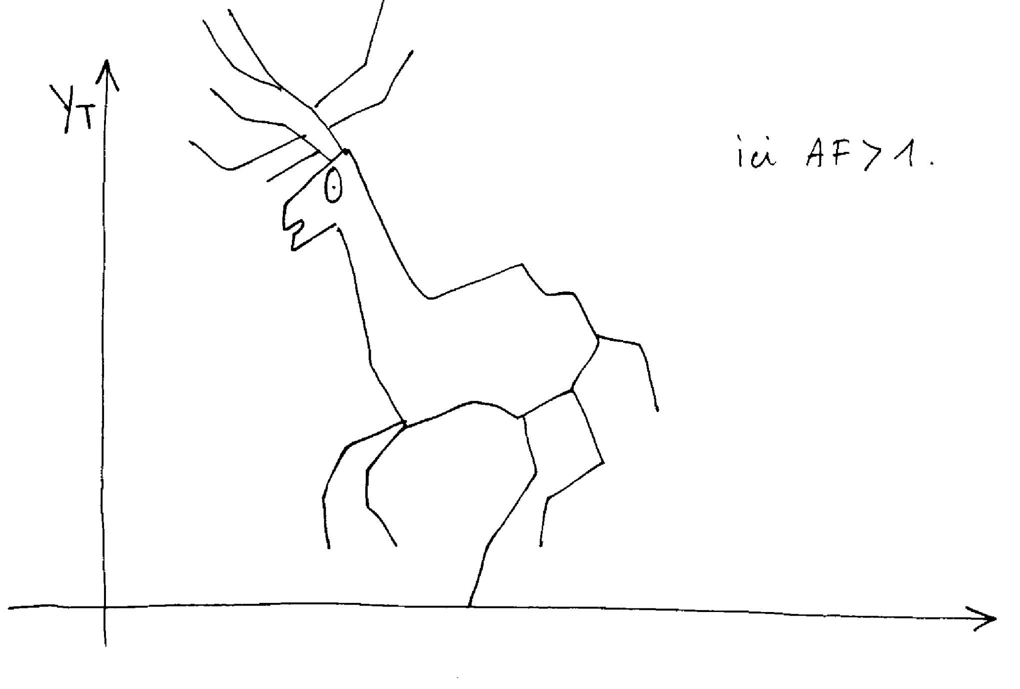
22170  NEXT J: NEXT I
22999  RETURN
```

AFFINITÉ

on peut envisager une affinité par rapport à un plan, ou par rapport à un axe. Donnons un exemple en deux dimensions XT, YT



Nous allons réaliser une affinité par exemple selon l'axe OXT
Appelons **AF** le **COEFFICIENT D'AFFINITÉ**.
Nous allons remplacer YT par AF * YT et nous obtiendrons le CERF-AFFIN :



On se limitera à l'affinité par rapport à un plan. Si besoin est, le lecteur pourra modifier le programme pour réaliser une affinité par rapport à un axe

```
23000 REM

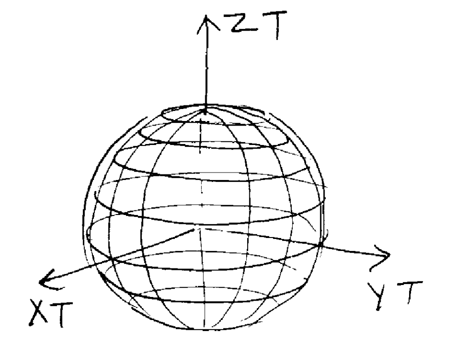
AFFINITE

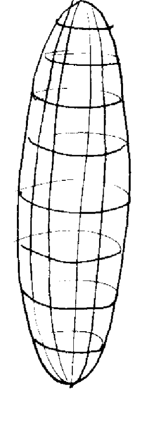
23010 HOME :
23015 FX = 0:FY = 0:FZ = 0
23020 GOSUB 30000
23060 PRINT "1- X=0 PLAN YOZ": PRINT
23070 PRINT "2- Y=0 PLAN XOZ": PRINT
23080 PRINT "3- Z=0 PLAN XOY": PRINT
23090 INPUT "VOTRE CHOIX ";C
23100 IF C = 1 THEN FX = 1
23110 IF C = 2 THEN FY = 1
23120 IF C = 3 THEN FZ = 1
23125 PRINT : INPUT "COEF AFFINITE";AF: PRINT
23130 FOR I = 1 TO L: FOR J = 1 TO N(I): REM

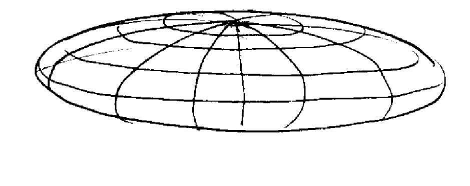
23140 IF FX = 1 THEN XT(I,J) = AF * XT(I,J)
23150 IF FY = 1 THEN YT(I,J) = AF * YT(I,J)
23160 IF FZ = 1 THEN ZT(I,J) = AF * ZT(I,J): REM

23170 NEXT J: NEXT I
23999 RETURN
```

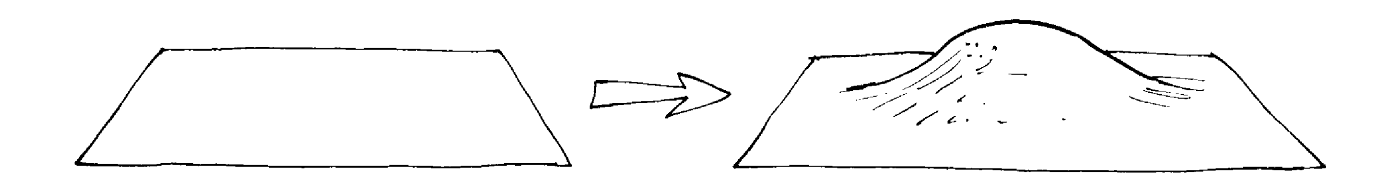
On voit donc qu'on peut disposer de tout un attirail de programmes qui permettent de manipuler des objets. Imaginons que nous ayions en mémoire un objet qui serait une sphère. En la déformant par affinité, il serait possible d'en faire un ellipsoïde, allongé ou aplati.



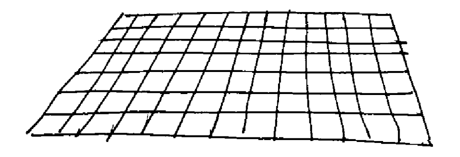


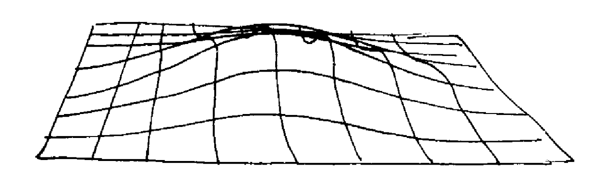


Continuons cette démarche de chaudronnerie. Comment créer une bosse dans une surface. Par exemple une surface plane

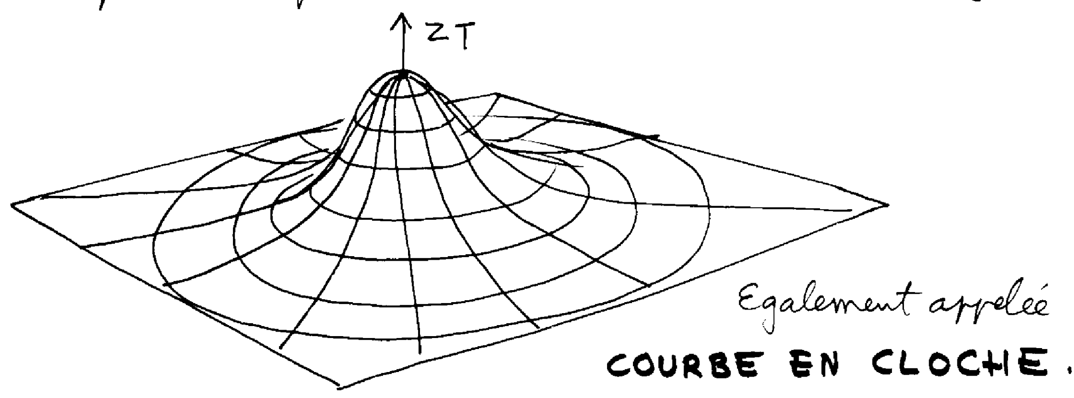


le matériau de départ pourra être cette GRILLE, ce morceau de grillage, que nous avons fabriqué tout à l'heure.





On pourrait envisager de le déformer comme ceci, en réalisant une **HOMOTOPIE**. On envisagera de réaliser cette déformation dans trois directions au choix (OXT, OYT, OZT). Supposons que cette grille soit dans le plan Z = 0. On va rajouter une **SURÉPAISSEUR** variable selon l'endroit. Cette déformation s'exercera selon OZT et sera maximale en O. Puis son importance ira, décroissant, au fur et à mesure que l'on s'éloigne de l'axe OZT. Disons que cette surépaisseur correspondra par exemple à une **SURFACE DE GAUSS** :



Egalement appelée **COURBE EN CLOCHE**.

Soit ρ la distance d'un point à l'axe OZT

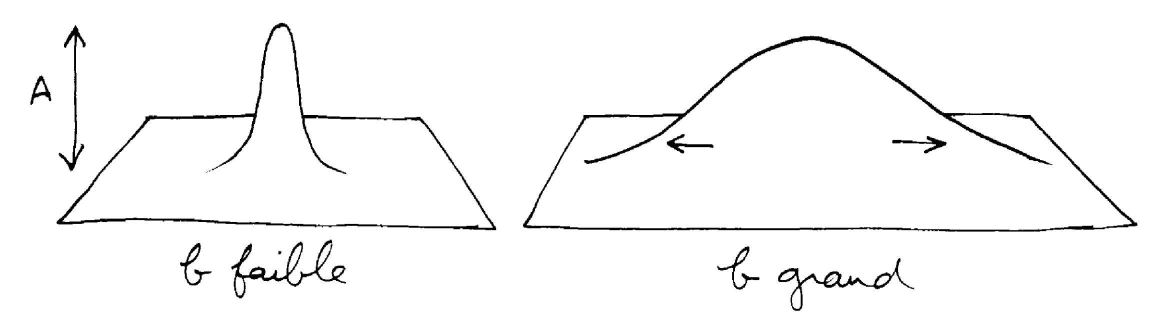
$$\rho = \sqrt{XT^2 + YT^2}$$

On introduira une déformation d'amplitude variable :

$$A(XT, YZ) = A \exp\left(-\rho^2 / 2b^2\right)$$

b sera quelquechose comme la largeur caractéristique de la bosse. Et A sera l'amplitude maximale (selon l'axe OZT).

Pour une même valeur de A, nous aurons :



le programme ci-après concrétise cette déformation :

```
24000  REM

HOMOTOPIE


24010 FX = 0:FY = 0:FZ = 0
24020  GOSUB 30000
24052  PRINT : INPUT "AMPLITUDE DEFORMATION ";A: PRIN
T
24054  INPUT "LARGEUR DEFORMATION ";B: PRINT
24060  PRINT "1-SELON OX": PRINT
24070  PRINT "2-SELON OY": PRINT
24080  PRINT "3-SELON OZ": PRINT
24090  INPUT "VOTRE CHOIX ";C: PRINT
24092  IF C = 1 THEN FX = 1
24093  IF C = 2 THEN FY = 1
24094  IF C = 3 THEN FZ = 1
24100  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

24110 X1 = XT(I,J):Y1 = YT(I,J):Z1 = ZT(I,J)
24140 R2 = (FX + FY) * Z1 * Z1 + (FX + FZ) * Y1 * Y1 + (FZ + FY
) * X1 * X
   1
24150 H = A * ( EXP ( - (R2 / (2 * B * B)))): REM

24160 XT(I,J) = XT(I,J) + FX * H
24170 YT(I,J) = YT(I,J) + FY * H
24180 ZT(I,J) = ZT(I,J) + FZ * H
24220  NEXT J: NEXT I
24999  RETURN
```

On pourrait, de même, créer des champs de déformations variés, dans l'espace. Par exemple des torsions. Imaginons une torsion selon l'axe OZT. TM serait l'amplitude maximale de cette torsion et B sa largeur caractéristique. Il suffirait de modifier les coordonnées XT(I,J), YT(I,J) en opérant une rotation dont l'amplitude irait en décroissant, au fur et à mesure que l'on s'éloignerait de l'axe OZT, par exemple selon un angle

$$T(\rho) = TM \exp -(\rho^2/2 b^2)$$

avec $\rho = \sqrt{XT^2 + YT^2}$

Le programme n'est pas difficile à imaginer. Il suffit de se servir de ce qui a été déjà donné.

FICHIERS BLOCS

Pour fabriquer des objets, il suffit d'utiliser les outils présentés ici et de fusionner les éléments crées et manipulés separément.

Mais la fusion a ses limites. On a vu que le programme n'était pas conçu pour manipuler des objets composés de plus de 300 "segments".

Que faire ?

Les objets sont destinés à être traités par un autre programme que nous appelerons **PANGRAPHE DESSIN** et qui ne pourra, lui aussi, gèrer que 300 segments **A' LA FOIS**.

On voit poindre la solution : **PANGRAPHE DESSIN** pourra gérer des ensembles d'objets, eux mêmes décrits dans des fichiers qu'on appelera **BLOCS D'OBJET** ou **FICHIERS BLOC.**

EL$(I) seront les éléments de ce fichier.

autrement dit des fichiers de fichiers...

```
28000 REM
CREATION D'UN FICHIER BLOC

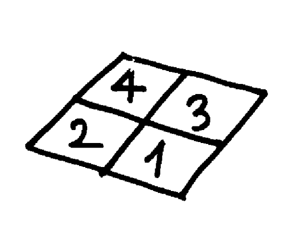
28010 HOME
28020 INPUT "NOMBRE D'OBJETS ";E
28030 FOR I = 1 TO E
28040 PRINT "OBJET NUMERO ";I
28050 INPUT EL$(I): PRINT
28060 NEXT I
28070 INPUT "NOM DE CE BLOC ";BL$: PRINT
28080 PRINT D$"OPEN";BL$
28090 PRINT D$"WRITE";BL$
28100 PRINT E
28110 FOR I = 1 TO E
28120 PRINT EL$(I)
28130 NEXT I
28140 PRINT D$"CLOSE";BL$
28999 RETURN
```

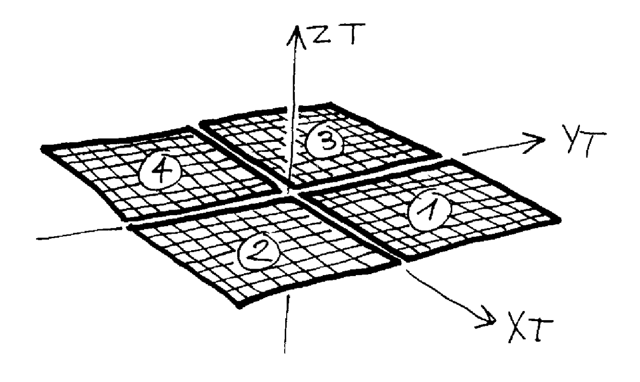
Et l'opération inverse :

```
29000 REM
LECTURE FICHIER BLOC

29005 HOME : INPUT "NOM DU BLOC ";BL$: PRINT
29010 D$ = CHR$ (13) + CHR$ (4)
29020 PRINT D$"OPEN";BL$
29030 PRINT D$"READ";BL$
29040 INPUT E
29050 FOR I = 1 TO E
29060 INPUT EL$(I)
29065 NEXT I
29070 PRINT D$"CLOSE";BL$
29080 FOR I = 1 TO E: PRINT EL$(I): NEXT : FOR TT = 1 TO 2
000: NEXT TT
29999 RETURN
```

Nous avons vu, pages 50, 51, 52 comment engendrer une grille, que nous appelerons GRILLE 1. En utilisant les sous options "symétrie par rapport à un plan, on engendredra aisément un sol pavé correspondant au dessin ci-après :





Cet ensemble pourra être géré par un fichier qu'on pourra appeler BLOC GRILLE (Ø/Ø/Ø) puisque son centre de gravité est à l'origine. Fichier qui sera constitué des quatre éléments

GRILLE1
GRILLE2
GRILLE3
GRILLE4

On notera que le point (Ø, Ø, Ø) n'apartient pas à cet ensemble ce qui permettra de "PASSER AU TRAVES" de ce bloc grille en suivant l'axe OZ.

La saisie et la manipulation d'objets peuvent comporter de nombreuses autres facettes. Mais nous arrêterons là notre exploration.

Ce programme ne permet présentement que de créer des objets et de jongler quelque peu avec. Il nous faut un programme de dessin.

Mais la section **PANGRAPHE OBJET** est déja si importante que nous n'aurons pas assez de place dans la mémoire de l'Apple II, dans nos 48K.

La solution va consister à placer dans PANGRAPHE OBJET quelquechose, des options liées aux IMAGES, au DESSIN, qui appeleront automatiquement PANGRAPHE DESSIN, également présent sur la diskette.

Dans notre menu nous avons fait figurer les options liées à PANGRAPHE DESSIN entre parenthèses. Il suffira que ces options renvoient à une séquence-transfert.

Remarque : peut-on conserver les fichiers en passant ainsi d'un programme à l'autre ? oui, à condition d'utiliser un programme-machine nommé **CHAIN.**

En utilisant le programme de copie **FID** présent sur la **MASTER**, vous pouvez dupliquer **CHAIN** sur votre diskette de travail. La séquence transfert sera alors :

```
40000 REM
CHAIN PANGRAPHE DESSIN

40010 PRINT D$"BLOADCHAIN,A520"
40020 CALL 520"PANGRAPHE DESSIN"
```

Il reste un sous programme de questionnement, souvent utilisé :

```
30000  REM
L'OBJET EST-IL EN MEMOIRE?

30010  PRINT : INPUT "L'OBJET EST-IL EN MEMOIRE ? ";R$:
 PRINT
30020  IF  LEFT$ (R$,1) = "O" THEN 30999
30030  INPUT "NOM DE L'OBJET: ";O$: PRINT
30040  GOSUB 4020
30999  RETURN
```

On comprend maintenant la présence de FLAGS dans le tout début du programme.

Si on commence par RUN PANGRAPHE OBJET 5/2/84 la ligne 2 assurera la création des places en mémoire pour les fichiers de points XT, YT, ZT. le flag FD sera alors positionné sur 1.

On pourra alors tourner dans PANGRAPHE OBJET. si on déclenche le transfert, le programme chainé PANGRAPHE DESSIN 5/2/84 aura les mêmes premières lignes. le Flag FD étant différent de zéro, on évitera une relecture de la ligne DIM, qui détruirait nos fichier L'option C ayant DEJA été définie, la ligne 6 évitera tout questionnement et 31Ø nous branchera automatiquement sur le bon sous programme.

Avant de construire PANGRAPHE DESSIN et de rendre tout cela plus explicite, donnons la suite complète du programme PANGRAPHE OBJET.

PANGRAPHE OBJET

```
0  IF FD = 1 THEN 5
1 L = 0: REM  PANGRAPHE OBJET 7/2/84
2  DIM XT(30,9),YT(30,9),ZT(30,9),X(30,9),Y(30,9),N(30):FD = 1
3 D$ =  CHR$ (13) +  CHR$ (4)
5  TEXT : HOME
6  IF C <  > 0 THEN 310
10  HOME
20  REM

 PROGRAMME MENU


30  VTAB 15: HTAB 15
40  PRINT "VOULEZ-VOUS:"
50  FOR TE = 1 TO 500: NEXT
60  HOME
70  PRINT
80  VTAB 3: HTAB 26: PRINT "PANGRAPHE OBJET"
90  VTAB 1
100  PRINT "1-CREER UN OBJET"
110  PRINT "2-STOCKER UN OBJET"
120  PRINT "3-CHARGER UN OBJET"
130  PRINT "4-COMPLETER UN OBJET"
140  PRINT "5-MODIFIER UN OBJET"
150  PRINT "6-(REPRESENTER UN OBJET)"
160  PRINT "7-(STOCKER UNE IMAGE)"
170  PRINT "8-(CHARGER UNE IMAGE)"
180  PRINT "9-MANIPULER UN OBJET"
190  PRINT "10-DEFINIR UN BLOC D'OBJETS"
200  PRINT "11-CONSULTER UN FICHIER BLOC"
210  PRINT "12-"
220  PRINT "13-"
230  PRINT "14-"
240  PRINT "15-"
250  PRINT "16-"
260  PRINT "17-AFFICHER LE CATALOGUE"
270  PRINT "18-SORTIR DU PROGRAMME"
280  INPUT "VOTRE CHOIX ";C
290  IF C = 17 THEN  PRINT D$"CATALOG": GET A$: GET A$
300  IF C = 18 THEN  END
310  ON C GOSUB 2000,3000,4000,2010,5000,40000,40000,4000
0,17000,28000,29
   000
999  GOTO 10
1000  REM

CREATION ARC DE CERCLE


1010  HOME :FX = 0:FY = 0:FZ = 0
1015  VTAB 15: HTAB 8
1020  HOME
1030  PRINT "COORDONNEES CENTRE DU CERCLE": PRINT
1040  INPUT "XC=";XC
1050  INPUT "YC=";YC
1060  INPUT "ZC=";ZC: PRINT
1070  PRINT "CERCLE D'AXE PARALLELE A:": PRINT
1080  PRINT "1-OX  2-OY  3-OZ": PRINT
1090  INPUT "VOTRE CHOIX ";C: PRINT
1095  INPUT "RAYON DU CERCLE ";R
1100  IF C = 1 THEN FX = 1
1110  IF C = 2 THEN FY = 1
1120  IF C = 3 THEN FZ = 1
1130  PRINT "SENS DE PARCOURS TRIGO HABITUEL": PRINT

1140  INPUT "DONNEZ THETA 1,DEGRES:";T1
1150  INPUT "DONNEZ THETA 2,DEGRES:";T2: PRINT
1160 T1 = T1 * 3.1416 / 180: REM  CONVERSION DEGRES/RA
DIANS
1170 T2 = T2 * 3.1416 / 180: REM  CONVERSION DEGRES/RA
DIANS
1180 DT = (T2 - T1) / 8
1190 L = L + 1: REM  NOUVELLE CHAINE
1200 N(L) = 9: REM  9 POINTS SUR LA CHAINE
1210  FOR K = 0 TO 8: REM  CREATION ARC DE CERCLE
1220 CT =  COS (T1 + DT * K):ST =  SIN (T1 + DT * K)
1230  REM LIGNES TRIGO DE THETA
1235 J = K + 1
```

→

```
1240 XT(L,J) = XC + (FY + FZ) * CT * R
1250 YT(L,J) = YC + (FX * CT + FZ * ST) * R
1260 ZT(L,J) = ZC + (FX + FY) * ST * R
1270  NEXT K
1275  FOR J = 1 TO 9
1280 XT(L,J) = ( INT (1000 * XT(L,J))) / 1000
1290 YT(L,J) = ( INT (1000 * YT(L,J))) / 1000
1300 ZT(L,J) = ( INT (1000 * ZT(L,J))) / 1000
1400  PRINT XT(L,J),YT(L,J),ZT(L,J)
1410  NEXT J
1420  GET A$
1999  RETURN
2000  REM
```

CREER UN OBJET

```
2005  HOME :L = 0
2010  HOME :: PRINT "VOULEZ-VOUS::": PRINT
2020  PRINT "1-CREER DES CHAINES"
2030  PRINT "2-POINTILLER DES SEGMENTS"
2040  PRINT "3-CREER UN CERCLE"
2050  PRINT "4-CHAINER DES CERCLES"
2060  PRINT "5-CHAINER DES COUPLES"
2070  PRINT "6-CREER UN ARC DE CERCLE": PRINT
2100  INPUT "VOTRE CHOIX ";C
2110  ON C GOSUB 10000,11000,12000,13000,14000,1000
2999  RETURN
3000  REM
```

STOCKER UN OBJET

```
3010  INPUT "NOM DE L'OBJET";O$
3020 D$ =  CHR$ (13) +  CHR$ (4)
3030  PRINT D$"OPEN";O$
3040  PRINT D$"WRITE";O$
3050  PRINT L
3060  FOR I = 1 TO L
3070  PRINT N(I)
3080  NEXT I
3090  FOR I = 1 TO L
3100  FOR J = 1 TO N(I)
3110  PRINT XT(I,J): PRINT YT(I,J): PRINT ZT(I,J)
3120  NEXT J: NEXT I
3130  PRINT D$"CLOSE";O$
3999  RETURN
4000  REM
```

CHARGEMENT OBJET

```
4010  INPUT "NOM DE L'OBJET ";O$
4020 D$ =  CHR$ (13) +  CHR$ (4)
4030  PRINT D$"OPEN";O$
4040  PRINT D$"READ";O$
4050  INPUT L
4060  FOR I = 1 TO L
4070  INPUT N(I)
4080  NEXT I
4090  FOR I = 1 TO L
4100  FOR J = 1 TO N(I)
4110  INPUT XT(I,J): INPUT YT(I,J): INPUT ZT(I,J)
4120  NEXT J: NEXT I
4130  PRINT D$"CLOSE";O$
4999  RETURN
5000  GOSUB 2000
5999  RETURN
10000  REM
```

CREATION DE CHAINES

```
10040 L = L + 1
10050  INPUT "NOMBRE DE POINTS? ";N(L)
10060 J = 0
10070 J = J + 1
10080  PRINT "POINT NO ";J
10090  INPUT "XT=";XT(L,J)
10100  INPUT "YT=";YT(L,J)
10110  INPUT "ZT=";ZT(L,J)
10120  IF J = N(L) THEN 10140
10130  GOTO 10070
10140  INPUT "UNE ERREUR?";R$
10150  IF  LEFT$ (R$,1) = "O" THEN 10050
10160  INPUT "UNE AUTRE CHAINE? ";R$
10170  IF  LEFT$ (R$,1) = "O" THEN 10040
10999  RETURN
```

```
11000 REM
```

POINTILLER UN SEGMENT

```
11010 L = L + 1:N(L) = 5
11020  PRINT "DEBUT DU SEGMENT "
11030  INPUT "X=";XD
11040  INPUT "Y=";YD
11050  INPUT "Z=";ZD
11060  PRINT "FIN DU SEGMENT"
11070  INPUT "X=";XF
11080  INPUT "Y=";YF
11090  INPUT "Z=";ZF
11100  FOR W = 1 TO 5
11110 XT(L,W) = XD + (XF - XD) * (W - 1) / 4
11120 YT(L,W) = YD + (YF - YD) * (W - 1) / 4
11130 ZT(L,W) = ZD + (ZF - ZD) * (W - 1) / 4
11140  NEXT W
11999  RETURN
12000  REM
```

CREATION D'UN CERCLE

```
12010  HOME
12015  PRINT "CENTRE DU CERCLE:": PRINT
12020  PRINT : INPUT "XC=";XC
12030  INPUT "YC=";YC
12040  INPUT "ZC=";ZC
12050  PRINT : INPUT "AZIMUT AXE ";AX
12060 AX = AX * 3.1416 / 180
12065  PRINT : INPUT "ANGLE DE SITE AXE:";SI
12066 SI = SI * 3.1416 / 180
12070  PRINT : INPUT "RAYON ";R
12080 CA =  COS (AX):SA =  SIN (AX)
12085 CS =  COS (SI):SS =  SIN (SI)
12090 L = L + 1
12100 N(L) = 9
12110 J = 0
12120 J = J + 1:W = (J - 1) * .785 / 2
12130 X0 = 0:Y0 =  - R *  COS (W):Z0 = R *  SIN (W)
12132 X1 =  - Z0 * SS + X0 * CS
12133 Y1 = Y0
12134 Z1 = Z0 * CS + X0 * SS
12140 XT(L,J) = X1 * CA - Y1 * SA + XC
12150 YT(L,J) = X1 * SA + Y1 * CA + YC
12160 ZT(L,J) = Z1 + ZC
12170  IF J = 9 THEN 12190
12180  GOTO 12120
12190 L = L + 1
12200 N(L) = 9
12210 J = 0
12220 J = J + 1:W = (J - 1) * .785 / 2
12230 X0 = 0:Y0 =  - R *  COS (W):Z0 =  - R *  SIN (W)
12232 X1 =  - Z0 * SS + X0 * CS
12233 Y1 = Y0
12234 Z1 = Z0 * CS + X0 * SS
12240 XT(L,J) = X1 * CA - Y1 * SA + XC
12250 YT(L,J) = X1 * SA + Y1 * CA + YC
12260 ZT(L,J) = Z1 + ZC
12270  IF J = 9 THEN 12999
12280  GOTO 12220
12999  RETURN
13000  REM
```

CHAINAGE DE CERCLES

```
13010  HOME : PRINT "CERCLES COAXIAUX // OX"
13012  PRINT : PRINT "AXE DES CERCLES": PRINT
13013  INPUT "YC=";YC
13014  INPUT "ZC=";ZC
13020 K = 0
13030 K = K + 1: PRINT : PRINT "CERCLE NO ";K
13040 L = L + 1:N(L) = 9
13050  INPUT "ABCISSE ";XC
13060  INPUT "RAYON ";R
13070  FOR J = 1 TO 9
13080 W = J * .785
13090 XT(L,J) = XC
13100 YT(L,J) = YC - R *  COS (W):ZT(L,J) = ZC + R *  SIN (W)
13110  NEXT J
13120  INPUT "UN AUTRE?";R$
13130  IF R$ = "OUI" THEN 13030
13140  FOR I = 1 TO 8
13150 L = L + 1:N(L) = K
13160  FOR J = 1 TO K
13170 XT(L,J) = XT(L - K - I + J,I)
13180 YT(L,J) = YT(L - K - I + J,I)
13190 ZT(L,J) = ZT(L - K - I + J,I)
13200  NEXT J
13210  NEXT I
13999  RETURN
```

```
14000  REM

CHAINER COUPLES


14010  PRINT "COUPLES AXES SUR OX"
14020 K = 0:L = 0
14030 K = K + 1:L = L + 1:N(L) = 5
14050  PRINT "COUPLE NUMERO ";K
14070  PRINT "DONNEZ 5 PTS :"
14080  FOR J = 1 TO 5
14090  PRINT "POINT NUMERO ";J
14100  INPUT "X=";XT(L,J)
14105  INPUT "Y=";YT(L,J)
14110  INPUT "Z=";ZT(L,J)
14120  NEXT J
14130  INPUT "UNE ERREUR? ";R$
14140  IF R$ = "OUI" THEN 14050
14230  INPUT "UN AUTRE COUPLE?";R$
14240  IF R$ = "OUI" THEN 14030
14260  FOR I = 1 TO 5
14270 L = L + 1:N(L) = K
14280  FOR J = 1 TO K
14290 XT(L,J) = XT(J,I)
14300 YT(L,J) = YT(J,I)
14310 ZT(L,J) = ZT(J,I)
14320  NEXT J: NEXT I
14331  FOR I = 1 TO K: FOR J = 1 TO 5
14332  PRINT XT(I,J),YT(I,J),ZT(I,J)
14333  NEXT J: NEXT I
14334  GET A$
14340  GOSUB 3000
14350  FOR I = 1 TO K + 5
14360  FOR J = 1 TO N(I)
14370 YT(I,J) =  - YT(I,J)
14380  NEXT J: NEXT I
14400  GOSUB 3000
14999  RETURN
17000  REM

MANIPULER UN OBJET


17010  HOME
17020  PRINT "VOULEZ-VOUS:": PRINT
17030  PRINT "1-OPERER UNE TRANSLATION DE L'OBJET":
PRINT
17040  PRINT "2-OPERER UNE ROTATION": PRINT
17050  PRINT "3-CREER LE SYMETRIQUE % UN PLAN": PRIN
T
17060  PRINT "4-FUSIONNER DEUX OBJETS": PRINT
17070  PRINT "5-HOMOTHETIE": PRINT
17080  PRINT "6-AFFINITE": PRINT
17090  PRINT "7-HOMOTOPIE": PRINT
17100  INPUT "VOTRE CHOIX ";C
17110  ON C GOSUB 19000,20000,18000,21000,22000,23000,240
00
17999  RETURN
18000  REM

SYMETRIQUE % PLAN


18001  HOME
18005 FX = 0:FY = 0:FZ = 0
18006  GOSUB 30000
18007  PRINT : PRINT "SYMETRIQUE PAR RAPPORT AU PLAN
": PRINT : PRINT "1-
   Z=0 PLAN XOY": PRINT : PRINT "2- Y=0 PLAN XOZ": PRINT
 : PRINT "3- X=
   0 PLAN YOZ": PRINT : INPUT "VOTRE CHOIX ";C
18008  IF C = 1 THEN FZ = 1
18009  IF C = 2 THEN FY = 1
18010  IF C = 3 THEN FX = 1
18011  FOR I = 1 TO L: FOR J = 1 TO N(I)
18020  IF FX = 1 THEN XT(I,J) =  - XT(I,J)
18022  IF FY = 1 THEN YT(I,J) =  - YT(I,J)
18024  IF FZ = 1 THEN ZT(I,J) =  - ZT(I,J)
18030  NEXT J: NEXT I
18035  PRINT
18040  INPUT "NOM NOUVEL OBJET ";O$
18050  GOSUB 3030
18809  IF C = 2 THEN FY = 1
18999  RETURN
```

```
19000  REM

TRANSLATION OBJET


19010  HOME
19020  GOSUB 30000
19040  PRINT "DONNEZ VECTEUR TRANSLATION": PRINT
19050  INPUT "DX=";DX
19060  INPUT "DY=";DY
19070  INPUT "DZ=";DZ
19080  FOR I = 1 TO L
19090  FOR J = 1 TO N(I)
19100 XT(I,J) = XT(I,J) + DX
19110 YT(I,J) = YT(I,J) + DY
19120 ZT(I,J) = ZT(I,J) + DZ
19130  NEXT J: NEXT I
19140  PRINT : PRINT "NOM OBJET TRANSLATE:": PRINT
19150  INPUT "";O$
19160  GOSUB 3030
19999  RETURN
20000  REM

ROTATION OBJET


20010  HOME
20020  GOSUB 30000
20025  PRINT
20030  PRINT "ROTATION AXE PARALLELE A:": PRINT
20040  PRINT "1-L'AXE OX : ROULIS": PRINT
20050  PRINT "2-L'AXE OY : TANGAGE": PRINT
20060  PRINT "3-L'AXE OZ : LACET": PRINT
20070  INPUT "VOTRE CHOIX ";C
20080  ON C GOSUB 25000,26000,27000
20999  RETURN
21000  REM

FUSION OBJETS


21010  HOME
21020  PRINT "FIN DE FUSION,TAPEZ LE MOT FIN": PRINT
21030  PRINT "PREMIER OBJET A FUSIONNER": PRINT
21035  GOSUB 30000
21040  INPUT "NOM OBJET A FUSIONNER:";O$
21042  IF O$ = "FIN" THEN 21500
21050  PRINT D$"OPEN";O$
21060  PRINT D$"READ";O$
21062  REM

OPERATION DE FUSION


21065  REM  LL=NOMBRE DE CHAINES  DU NOUVEL OBJET
21070  INPUT LL
21080  FOR I = L + 1 TO L + LL
21090  INPUT N(I)
21100  NEXT I
21110  FOR I = L + 1 TO L + LL
21120  FOR J = 1 TO N(I)
21130  INPUT XT(I,J): INPUT YT(I,J): INPUT ZT(I,J)
21140  NEXT J: NEXT I
21150  PRINT D$"CLOSE";O$
21155 L = L + LL
21160  GOTO 21040
21500  REM  FIN DE FUSION
21510  HOME
21520  PRINT : INPUT "NOM DE L'OBJET GLOBAL ";O$
21530  GOSUB 3030
21999  RETURN
22000  REM

HOMOTHETIE


22010  HOME
22020  GOSUB 30000
22040  PRINT "COORDONNEES CENTRE HOMOTHETIE": PRIN
T
22050  INPUT "XC=";XC
22060  INPUT "YC=";YC
22070  INPUT "ZC=";ZC
22080  PRINT : INPUT "COEFFICIENT HOMOTHETIE ";HO: PR
INT
22090  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

22100 X1 = XT(I,J) - XC
22110 Y1 = YT(I,J) - YC
22120 Z1 = ZT(I,J) - ZC: REM
```

→

```
22130 X2 = X1 * HO
22135 Y2 = Y1 * HO
22140 Z2 = Z1 * HO: REM

22145 XT(I,J) = X2 + XC
22150 YT(I,J) = Y2 + YC
22160 ZT(I,J) = Z2 + ZC: REM

22170  NEXT J: NEXT I
22999  RETURN
23000  REM

AFFINITE


23010  HOME :
23015 FX = 0:FY = 0:FZ = 0
23020  GOSUB 30000
23060  PRINT "1- X=0 PLAN YOZ": PRINT
23070  PRINT "2- Y=0 PLAN XOZ": PRINT
23080  PRINT "3- Z=0 PLAN XOY": PRINT
23090  INPUT "VOTRE CHOIX ";C
23100  IF C = 1 THEN FX = 1
23110  IF C = 2 THEN FY = 1
23120  IF C = 3 THEN FZ = 1
23125  PRINT : INPUT "COEF AFFINITE";AF: PRINT
23130  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

23140  IF FX = 1 THEN XT(I,J) = AF * XT(I,J)
23150  IF FY = 1 THEN YT(I,J) = AF * YT(I,J)
23160  IF FZ = 1 THEN ZT(I,J) = AF * ZT(I,J): REM

23170  NEXT J: NEXT I
23999  RETURN
24000  REM

HOMOTOPIE


24010 FX = 0:FY = 0:FZ = 0
24020  GOSUB 30000
24052  PRINT : INPUT "AMPLITUDE DEFORMATION ";A: PRIN
T
24054  INPUT "LARGEUR DEFORMATION ";B: PRINT
24060  PRINT "1-SELON OX": PRINT
24070  PRINT "2-SELON OY": PRINT
24080  PRINT "3-SELON OZ": PRINT
24090  INPUT "VOTRE CHOIX ";C: PRINT
24092  IF C = 1 THEN FX = 1
24093  IF C = 2 THEN FY = 1
24094  IF C = 3 THEN FZ = 1
24100  FOR I = 1 TO L: FOR J = 1 TO N(I): REM

24110 X1 = XT(I,J):Y1 = YT(I,J):Z1 = ZT(I,J)
24140 R2 = (FX + FY) * Z1 * Z1 + (FX + FZ) * Y1 * Y1 + (FZ + FY
) * X1 * X
    1
24150 H = A * ( EXP ( - (R2 / (2 * B * B)))): REM

24160 XT(I,J) = XT(I,J) + FX * H
24170 YT(I,J) = YT(I,J) + FY * H
24180 ZT(I,J) = ZT(I,J) + FZ * H
24220  NEXT J: NEXT I
24999  RETURN
25000  REM

ROTATION AXE // OX


25010  HOME
25020  PRINT "TRACE AXE DANS PLAN YOZ": PRINT
25030  INPUT "YA=";YA: PRINT : INPUT "ZA=";ZA: PRINT
25040  INPUT "ANGLE,DEGRES: ";AR
25050  REM  AR ANGLE DE ROULIS
25060 AR = AR * 3.1416 / 180
25070 CR =  COS (AR):SR =  SIN (AR)
25080  FOR I = 1 TO L: FOR J = 1 TO N(I)
25090 X1 = XT(I,J)
25100 Y1 = YT(I,J) - YA
25110 Z1 = ZT(I,J) - ZA: REM

25120 X2 = X1
25130 Y2 = Y1 * CR - Z1 * SR
25140 Z2 = Y1 * SR + Z1 * CR: REM
25150 XT(I,J) = X2
25160 YT(I,J) = Y2 + YA
25170 ZT(I,J) = Z2 + ZA: REM

25180  NEXT J: NEXT I
25999  RETURN
```

```
26000 REM

ROTATION AXE // OY


26010 HOME
26020 PRINT "TRACE AXE DANS PLAN XOZ": PRINT
26030 INPUT "XA=";XA: PRINT : INPUT "ZA=";ZA: PRINT
26040 INPUT "ANGLE DEGRES ";A1
26050 A1 = A1 * 3.1416 / 180
26070 REM  A1 ANGLE DE TANGAGE
26075 CT = COS (A1):ST = SIN (A1)
26080 FOR I = 1 TO L: FOR J = 1 TO N(I): REM

26100 X1 = XT(I,J) - XA
26110 Y1 = YT(I,J)
26120 Z1 = ZT(I,J) - ZA: REM

26140 X2 = Z1 * ST + X1 * CT
26150 Y2 = Y1
26160 Z2 = Z1 * CT - X1 * ST: REM
26170 XT(I,J) = X2 + XA
26180 YT(I,J) = Y2
26190 ZT(I,J) = Z2 + ZA: REM

26200 NEXT J: NEXT I
26999 RETURN
27000 REM

ROTATION AXE // OZ


27010 HOME
27020 PRINT "TRACE AXE DANS PLAN XOY": PRINT
27030 INPUT "XA=";XA: PRINT : INPUT "YA=";YA: PRINT
27040 INPUT "ANGLE,DEGRES: ";AL
27050 AL = AL * 3.1516 / 180
27060 REM  AL ANGLE DE LACET
27070 CL = COS (AL):SL = SIN (AL)
27080 FOR I = 1 TO L: FOR J = 1 TO N(I): REM

27090 X1 = XT(I,J) - XA
27100 Y1 = YT(I,J) - YA
27110 Z1 = ZT(I,J): REM

27120 X2 = X1 * CL - Y1 * SL
27130 Y2 = X1 * SL + Y1 * CL
27140 Z2 = Z1: REM

27150 XT(I,J) = X2 + XA
27160 YT(I,J) = Y2 + YA
27170 ZT(I,J) = Z2: REM

27180 NEXT J: NEXT I
27999 RETURN
28000 REM

CREATION D'UN FICHIER BLOC


28010 HOME
28020 INPUT "NOMBRE D'OBJETS ";E
28030 FOR I = 1 TO E
28040 PRINT "OBJET NUMERO ";I
28050 INPUT EL$(I): PRINT
28060 NEXT I
28070 INPUT "NOM DE CE BLOC ";BL$: PRINT
28080 PRINT D$"OPEN";BL$
28090 PRINT D$"WRITE";BL$
28100 PRINT E
28110 FOR I = 1 TO E
28120 PRINT EL$(I)
28130 NEXT I
28140 PRINT D$"CLOSE";BL$
28999 RETURN
29000 REM
LECTURE FICHIER BLOC


29005 HOME : INPUT "NOM DU BLOC ";BL$: PRINT
29010 D$ = CHR$ (13) + CHR$ (4)
29020 PRINT D$"OPEN";BL$
29030 PRINT D$"READ";BL$
29040 INPUT E
29050 FOR I = 1 TO E
29060 INPUT EL$(I)
29065 NEXT I
29070 PRINT D$"CLOSE";BL$
29080 FOR I = 1 TO E: PRINT EL$(I): NEXT : FOR TT = 1 TO 2
000: NEXT TT
29999 RETURN
```

```
30000 REM

L'OBJET EST-IL EN MEMOIRE?


30010 PRINT : INPUT "L'OBJET EST-IL EN MEMOIRE ? ";R$:
 PRINT
30020 IF  LEFT$ (R$,1) = "O" THEN 30999
30030 INPUT "NOM DE L'OBJET: ";O$: PRINT
30040 GOSUB 4020
30999 RETURN
40000 REM

CHAIN PANGRAPHE DESSIN


40010 PRINT D$"BLOADCHAIN,A520"
40020 CALL 520"PANGRAPHE DESSIN"


5000 REM

MODIFIER UN OBJET


5010 HOME
5020 GOSUB 30000
5030 PRINT : PRINT "OBJET DE ";L;" CHAINES": PRINT
5040 INPUT "DONNEZ L'INDICE DE CHAINE ";I
5050 INPUT "DONNEZ INDICE POINT SUR LA CHAINE ";J
5060 PRINT
5070 INPUT "XT(I,J)=";XT(I,J): PRINT
5080 INPUT "YT(I,J=";YT(I,J): PRINT
5090 INPUT "ZT(I,J)=";ZT(I,J): PRINT
5100 INPUT "UN AUTRE POINT ";R$
5110 IF  LEFT$ (R$,1) = "O" THEN 5040
5120 GOSUB 3020
5998 END
```

PANGRAPHE DESSIN

Cette section représente la deuxième partie de cet ensemble.
le Menu de **PANGRAPHE** sera semblable à celui de **PANGRAPHE OBJET**:

1-(CREER UN OBJET)
2-(STOCKER UN OBJET
3-CHARGER UN OBJET PANGRAPHE DESSIN
4-(COMPLETER UN OBJET
5-(MODIFIER UN OBJET)
6-REPRÉSENTER UN OBJET
7-STOCKER UNE IMAGE
8-CHARGER UNE IMAGE
9-(MANIPULER UN OBJET)
1Ø-(DEFINIR UN BLOC D'OBJETS)
11-CONSULTER UN FICHIER BLOC
12-
13-
14-
15-
16-
17-AFFICHER LE CATALOGUE
18-SORTIR DU PROGRAMME

VOTRE CHOIX ? ■

Comme on peut le voir dans ce menu, les parenthèses sont " inversées". Elle encadreront les options appartenant à PANGRAPHE OBJET.

```
]LOADPANGRAPHE DESSIN
]LIST

0  IF FD = 1 THEN 5
1 L = 0: REM PANGRAPHE DESSIN 5/2/84
2  DIM XT(30,9),YT(30,9),ZT(30,9),X(30,9),Y(30,9),N(30):FD = 1
3 D$ =  CHR$ (13) +  CHR$ (4)
5  TEXT : HOME
6  IF C <  > 0 THEN 330
10  HOME
20  REM

 PROGRAMME MENU


30  VTAB 15: HTAB 15
40  PRINT "VOULEZ-VOUS:"
50  FOR TE = 1 TO 500: NEXT
60  HOME
70  PRINT
80  VTAB 3: HTAB 25: PRINT "PANGRAPHE DESSIN"
90  VTAB 1
100  PRINT "1-(CREER UN OBJET)"
110  PRINT "2-(STOCKER UN OBJET)"
120  PRINT "3-CHARGER UN OBJET"
130  PRINT "4-(COMPLETER UN OBJET)"
140  PRINT "5-(MODIFIER UN OBJET)
150  PRINT "6-REPRESENTER UN OBJET"
160  PRINT "7-STOCKER UNE IMAGE"
170  PRINT "8-CHARGER UNE IMAGE"
180  PRINT "9-(MANIPULER UN OBJET)"
190  PRINT "10-(DEFINIR UN BLOC D'OBJETS)"
200  PRINT "11-CONSULTER UN FICHIER BLOC"
210  PRINT "12-"
220  PRINT "13-"
230  PRINT "14-"
240  PRINT "15-"
250  PRINT "16-"
260  PRINT "17-AFFICHER LE CATALOGUE"
270  PRINT "18-SORTIR DU PROGRAMME"
290  INPUT "VOTRE CHOIX ";C
300  IF C = 17 THEN  PRINT D$"CATALOG": GET A$: GET A$
310  IF C = 18 THEN  END
320  IF C = 19 THEN  PRINT D$"RUNPANGRAPHE PRINCIPAL
"
330  ON C GOSUB 40000,40000,4000,40000,40000,7000,8000,90
00,40000,40000,2
   9000
999  GOTO 10
```

Au début, même système de FLAG FD, même test sur la valeur de C.

Les options absentes, renvoient à la routine de transfert 40 000, encore présente :

```
40000  REM

CHAINAGE SUR P/OBJET
40010  PRINT D$"BLOADCHAIN,A520"
40020  CALL 520"PANGRAPHE OBJET"
```

Pour dessiner un objet, il faudra commencer par le charger en mémoire centrale. Ce qui fait que l'on trouvera encore ici la même routine de chargement :

```
4000  REM

CHARGEMENT OBJET

4010  INPUT "NOM DE L'OBJET ";O$
4020 D$ =  CHR$ (4)
4030  PRINT D$"OPEN";O$
4040  PRINT D$"READ";O$
4050  INPUT L
4060  FOR I = 1 TO L
4070  INPUT N(I)
4080  NEXT I
4090  FOR I = 1 TO L
4100  FOR J = 1 TO N(I)
4110  INPUT XT(I,J): INPUT YT(I,J): INPUT ZT(I,J)
4120  NEXT J: NEXT I
4130  PRINT D$"CLOSE";O$
4999  RETURN
5000  GOSUB 2000
5999  RETURN
```

Au rayon des choses déjà connues :

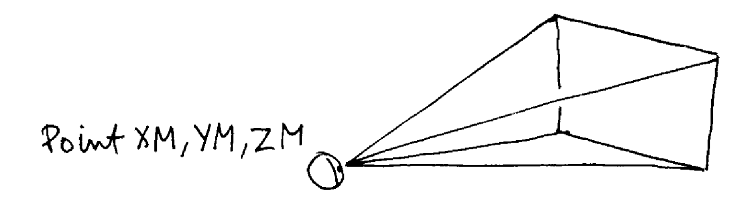
```
29000  REM
LECTURE FICHIER BLOC

29005  HOME : INPUT "NOM DU BLOC ";BL$: PRINT
29010 D$ =  CHR$ (4)
29020  PRINT D$"OPEN";BL$
29030  PRINT D$"READ";BL$
29040  INPUT E
29050  FOR I = 1 TO E
29060  INPUT EL$(I)
29065  NEXT I
29070  PRINT D$"CLOSE";BL$
29080  FOR I = 1 TO E: PRINT EL$(I): NEXT I
29090  FOR TT = 1 TO 2000: NEXT TT
29999  RETURN
```

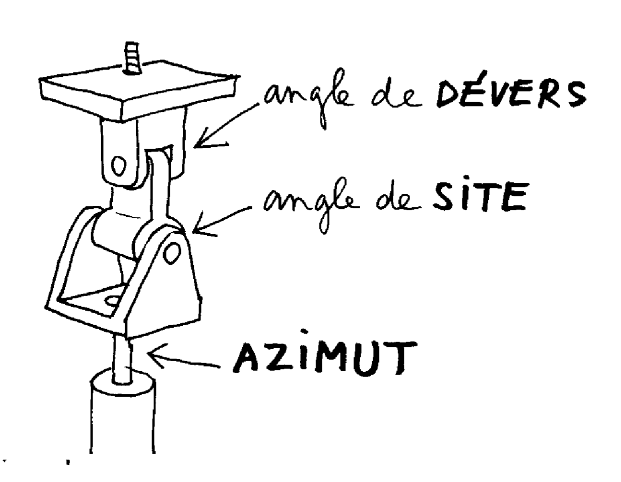
Tout le reste concerne la fabrication d'Images.

REPRÉSENTER UN OBJET

l'objet (ou les objets) que nous voulons représenter, se situent quelquepart dans l'espace objet XT, YT, ZT.
Pour créer une vue en perspective il va falloir indiquer un **POINT D'OBSERVATION** de coordonnées XM, YM, ZM.
Il faut ensuite indiquer à la machine dans quelle direction il lui faut "regarder" cet objet, c'est à dire orienter la **FENÊTRE DE VISION** :



Nous savons que nous pouvons à volonté régler l'**OUVERTURE ANGULAIRE AN** de cette fenêtre, c'est à dire notre **FOCALE**, notre **ZOOM**.
l'**ATTITUDE** de l'observateur dépend de trois angles, que l'on retrouve sur un pied d'appareil photographique.



Pour être tout à fait complet, on devrait les faire intervenir tous les trois. Mais, dans cette première version de **PANGRAPHE** nous ne retiendrons que l'**AZIMUT** et le **SITE**. On a en général pas l'habitude de regarder un objet en se tordant le cou de côté. C'est mauvais pour les vertèbres, et on risque d'attrapper un torticolis.



L'angle de dévers devra être ultérieurement introduit si on souhaite par exemple représenter sur l'écran ce que voit un pilote pendant un virage (simulateur de pilotage

merde, la tour !..

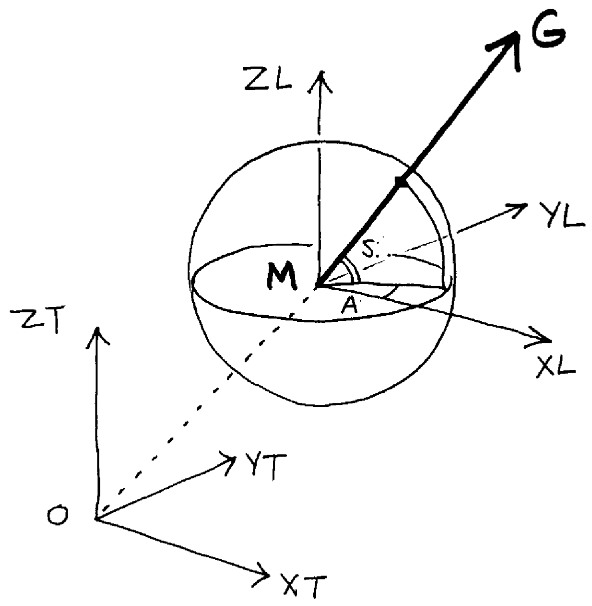
DÉVERS DU AU ROULIS

LACET

ROULIS

TANGAGE

La première idée consisterait à indiquer à l'ordinateur, soi-même, les angles d'azimut et de site. Mais cela impose un calcul pénible, que l'on peut tout aussi bien confier à la machine.



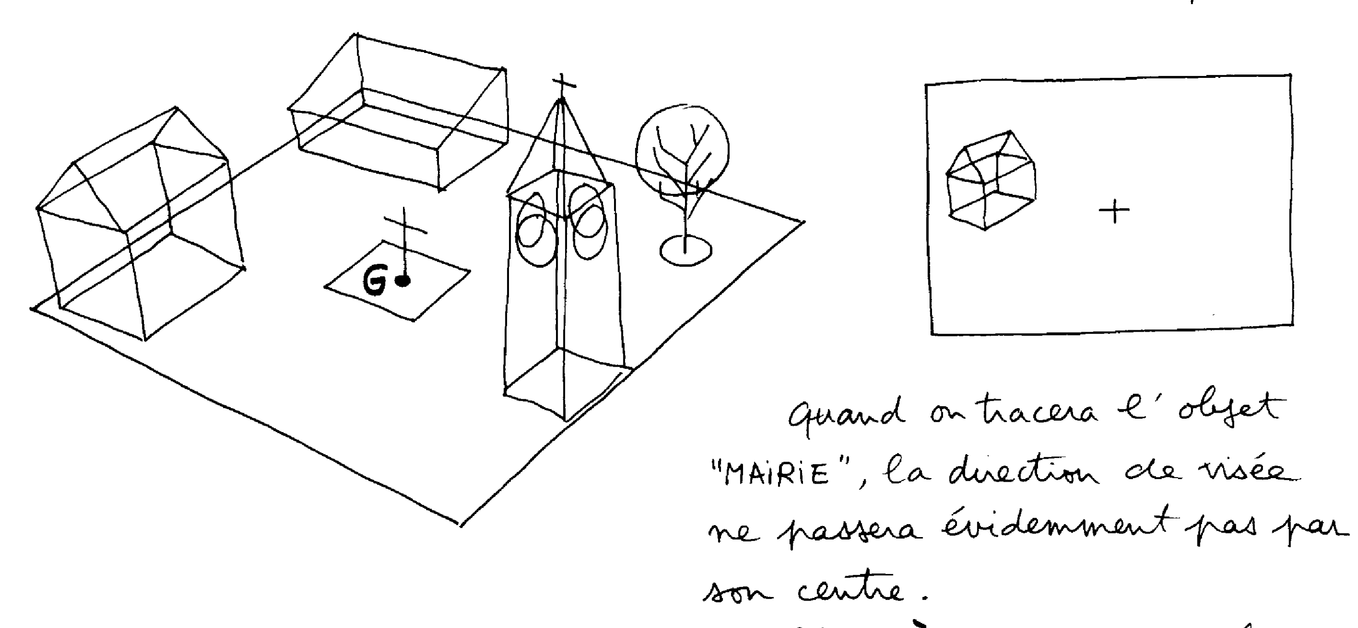
$(\overrightarrow{OX_T}, \overrightarrow{OY_T}, \overrightarrow{OZ_T})$ représente le **RÉFÉRENTIEL ABSOLU**, vis à vis duquel on définit les coordonnées (XT, YT, ZT) des **POINTS OBJETS**.

M est le point d'observation, de coordonnées (XM, YM, ZM) et le vecteur **MG** la direction de pointage. On va faire intervenir un trièdre trirectangle $(\overrightarrow{MXL}, \overrightarrow{MYL}, \overrightarrow{MZL})$ centré en M et dont les axes sont parallèles aux axes du trièdre $(\overrightarrow{OXT}, \overrightarrow{OYT}, \overrightarrow{OZT})$.

Sur cette figure, on voit très bien les angles **A** (Azimut) et **S** (site).

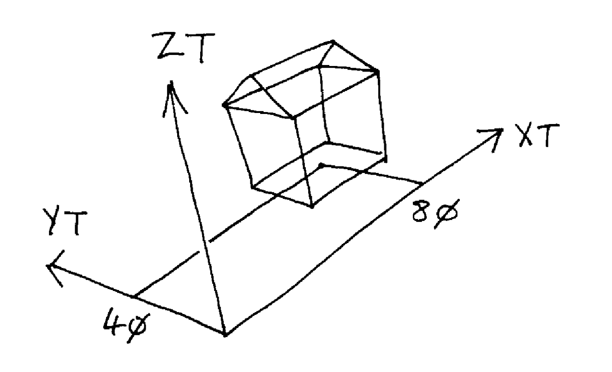
La visée se fait en direction d'un point **G** (XG, YG, ZG), et c'est cet ensemble de coordonnées qui sera fourni à la machine. Si le point **G** se trouve être le centre de gravité de l'objet, par exemple, on aura toutes les chances de regarder dans la bonne direction.

A ce stade, on pourrait se dire : pourquoi ne pas inclure ces coordonnées objet XG, YG, ZG dans le fichier OBJET, et stocker cette informatique. Lors du dessin, le pointage pourrait se faire automatiquement. Mais cela serait trop restrictif et trop rigide. Si nous voulons par exemple représenter une place de village, avec plusieurs objets-maison : le point de visée pourra être le centre de la place.



quand on tracera l'objet "MAIRIE", la direction de visée ne passera évidemment pas par son centre.

Le problème est surtout de ne pas oublier **OÙ** se trouve un objet mémorisé sur la diskette. Sinon on risque de perdre pas mal de temps à le rechercher. Une solution consiste à faire figurer les coordonnées de son centre dans le nom de fichier.

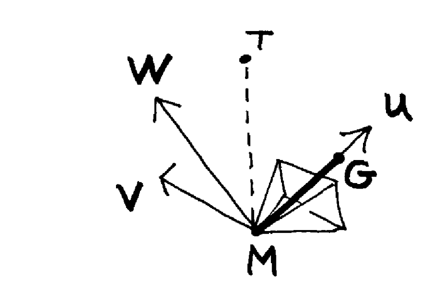


Exemple : cet objet s'appelera MAIRIE8Ø-4Ø-Ø ou mieux : MAIRIE(8Ø/4Ø/Ø)

le vecteur $\overrightarrow{MG}$ a pour composantes

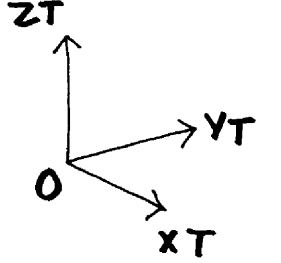
$$\begin{cases} CX = XG - XM \\ CY = YG - YM \\ CZ = ZG - ZM \end{cases}$$

On connait la suite d'instructions permettant de calcul l'azimut A et le site S. Il faut introduire maintenant un troisième trièdre **LIÉ À L'OBSERVATEUR :**



L'axe $\overrightarrow{MU}$ est colinéaire à $\overrightarrow{MG}$.

l'axe $\overrightarrow{MV}$ est horizontal
l'axe $\overrightarrow{MW}$ complète ce trièdre trirectangle



si **T** est un point de l'objet, ses coordonnées "apparentes" par rapport à ce nouveau trièdre, seront (**XA**, **YA**, **ZA**)

a partir de ces coordonnées apparentes nous pourront calculer l'azimut et le site "apparents", qui deviendront les coordonnées écran.

On aura :

$$\begin{cases} XA = \overrightarrow{MT} \cdot \overrightarrow{MU} \\ YA = \overrightarrow{MT} \cdot \overrightarrow{MV} \\ ZA = \overrightarrow{MT} \cdot \overrightarrow{MW} \end{cases}$$

avec $\overrightarrow{MT} \begin{cases} XT - XM \\ YT - YM \\ ZT - ZM \end{cases}$

$$\vec{MU}\begin{cases} XU = \cos S \cos A \\ YU = \cos S \sin A \\ ZU = \sin S \end{cases} \quad \vec{MV}\begin{cases} XV = -\sin A \\ YV = \cos A \\ ZV = 0 \end{cases} \quad MW\begin{cases} XW = -\sin S \cos A \\ YW = -\sin S \sin A \\ ZW = \cos S \end{cases}$$

On posera : $\begin{cases} \cos A = CT \\ \cos S = CK \end{cases} \quad \begin{cases} \sin A = ST \\ \sin S = SK \end{cases}$

d'ou

$$\vec{MU}\begin{cases} XU = CK * CT \\ YU = CK * ST \\ ZU = SK \end{cases} \quad \vec{MV}\begin{cases} XV = -ST \\ YV = CT \\ ZV = 0 \end{cases} \quad \vec{MW}\begin{cases} XW = -SK * CT \\ YW = -SK * ST \\ ZW = CK \end{cases}$$

$$\begin{cases} XA = XL * XU + YL * YU + ZL * ZU \\ YA = XL * XV + YL * YV + ZL * ZV \\ ZA = XL * XW + YL * YW + ZL * ZW \end{cases}$$

produits scalaires donnant les **COORDONNÉES APPARENTES.**

Le programme suivant va matérialiser ces formules.

```
7000  REM

REPRESENTER UN OBJET


7010  TEXT : HOME
7020  PRINT : PRINT
7030  INPUT "OUVERTURE ANGULAIRE ";AN: PRINT
7040  PRINT "POINT D'OBSERVATION"
7050  PRINT : INPUT "X=";XM
7060  INPUT "Y=";YM
7070  INPUT "Z=";ZM
7080  PRINT : PRINT "CENTRE G DE L'IMAGE": PRINT : INPU
T "XG=";XG: INPUT
   "YG=";YG: INPUT "ZG=";ZG
7090 CX = XG - XM:CY = YG - YM:CZ = ZG - ZM
7100 DI =  SQR (CX * CX + CY * CY)
7110  IF DI = 0 AND CZ > 0 THEN TE = 0:KI = 3.1416 / 2: GOTO
 7210
7120  IF DI = 0 AND CZ < 0 THEN TE = 0:KI =  - 3.1416 / 2: GO
TO 7210
7130 KI =  ATN (CZ / DI)
7140  IF CX = 0 AND CY > 0 THEN TE = 1.57: GOTO 7210
7150  IF CX = 0 AND CY < 0 THEN TE =  - 1.57: GOTO 7210
7160 TE =  ATN (CY / CX)
7170  IF CX < 0 AND CY > 0 THEN TE = 3.1416 + TE
7180  IF CX < 0 AND CY < 0 THEN TE = 3.1416 + TE
7190  IF CX < 0 AND CY = 0 THEN TE = 3.1416
7200  IF CX = 0 AND CY < 0 THEN TE =  - 3.1416 / 2
7210 AZ = TE * 170 / 3.1416:SI = KI * 170 / 3.1416
7220  REM  AZ AZIMUT;SI SITE
7230 CT =  COS (TE):ST =  SIN (TE)
7240 CK =  COS (KI):SK =  SIN (KI)
7250 XU = CK * CT:YU = CK * ST:ZU = SK
7260 XV =  - ST:YV = CT:ZV = 0
7270 XW =  - SK * CT:YW =  - SK * ST:ZW = CK
7280  HOME
7290  PRINT "1-OBJET DEJA EN MEMOIRE": PRINT
7300  PRINT "2-DESSIN PAR ELEMENTS": PRINT
7310  PRINT "3-DESSIN PAS A PAS": PRINT
7320  PRINT "4-DESSIN PAR BLOC": PRINT
7330  INPUT "VOTRE CHOIX ";C: PRINT
7340  ON C GOSUB 12000,11000,30000,10000
7998  GET A$: TEXT
7999  RETURN
```

De nouveau, un sous menu et des options

Option 1 : l'objet est DÉJA en mémoire. Un dessin d'un objet UNIQUE, qu'on aura pas besoin de charger.

Option 2 : Dessin par éléments.

On indiquera à l'ordinateur que l'on souhaite le tracé d'un décor comprenant un certain nombre d'objets, qui seront saisis par constitution d'un fichier d'éléments EL$(I) qui seront ensuite chargés un à un et dessinés

Option 3 : Dessin pas à pas : dans une alternance de pages graphiques et de pages texte, on pourra surimpressionner, pas à pas, de nouveaux objets

Option 4 : Dessin par bloc : l'ensemble d'éléments a préalablement été défini et forme un fichier-bloc présent sur la diskette.

le sous programme de tracé d'image, à partir des données visuelles et d'un objet présent en mémoire correspondra au sous programme 15ØØØ. On va le garder pour la fin.

OBJET DEJA EN MÉMOIRE :

```
12000  REM
OBJET DEJA EN MEMOIRE

12010  HGR2 : HCOLOR= 3
12020  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
12030  GOSUB 15000
12040  FOR T = 1 TO 7: PRINT  CHR$ (7): NEXT T
12999  RETURN
```

En prime : on trace le cadre ...

DESSIN PAR ÉLÉMENTS :

```
11000  REM
DESSIN PAR ELEMENTS

11010  HOME
11020  INPUT "NOMBRE D'ELEMENTS ";E
11030  FOR K = 1 TO E
11040  INPUT "NOM ELEMENT ";EL$(K)
11045  NEXT K
11050  HGR2 : HCOLOR= 3
11060  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
11070  FOR K = 1 TO E
11080 O$ = EL$(K)
11090  GOSUB 4020
11100  GOSUB 15000
11110  NEXT K
11120  FOR T = 1 TO 7: PRINT  CHR$ (7): NEXT T
11999  RETURN
```

Première partie : Saisie des éléments constituant le fichier EL$ (K)
Deuxième partie : tracé

DESSIN PAS À PAS

```
30000  REM
DESSIN PAS A PAS

30010  HOME
30020  INPUT "NOM PREMIER ELEMENT ";O$: PRINT
30030  GOSUB 4020
30040  HGR2 : HCOLOR= 3
30050  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
30060  GOSUB 15000: FOR TT = 1 TO 7: PRINT  CHR$ (7): NEXT
 TT
30070  GET A$: GET A$
```

Première partie : on charge le premier élément. On trace le cadre, puis l'image.
Suite de sept "BIPS" sonores (CHR$(7)).
Puis pause à l'aide de GET A$
Une suite de **POKE** permet alors de retourner en mode Texte SANS détruire l'image, qui sera tracée en PAGE II et qui restera intacte.

En mode texte, questionnement :
UN AUTRE ELEMENT ?

si c'est non, fin de ce sous-programme, par saut en 30999 RETURN
si c'est oui, nom de l'élément, chargement, retour en mode graphique par des POKES AD HOC et dessin (GOSUB 15000), et nouvelle pause.

```
30080  POKE  - 16303,0: POKE  - 16300,0: REM TEXTE
30090  INPUT "UN AUTRE ELEMENT ? ";R$: PRINT
30100  IF  LEFT$ (R$,1) <  > "O" THEN 30999
30110  INPUT "NOM DE L'ELEMENT ";O$: PRINT
30120  GOSUB 4020
30130  POKE  - 16304,0: POKE  - 16299,0: REM PAGEII
30140  GOSUB 15000: FOR TT = 1 TO 7: PRINT  CHR$ (7): NEXT
 TT
30150  GET A$: GET A$
30160  GOTO 30080
30999  RETURN
```

DESSIN PAR BLOC :

```
10000  REM

DESSIN PAR BLOC

10010  INPUT "NOM DU BLOC ";BL$: PRINT
10030  HGR2 : HCOLOR= 3
10040  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
10050  FOR K = 1 TO E
10060 O$ = EL$(K)
10070  GOSUB 4020
10080  GOSUB 15000
10090  NEXT K
10100  FOR T = 1 TO 7: PRINT  CHR$ (7): NEXT T
10999  RETURN
```

après avoir saisi le nom du Fichier Bloc, l'ordinateur le charge (GOSUB 29000).
Puis tout se déroule de manière très similaire

On ne sait toujours pas comment on crée un dessin.
Mais on peut donner les sous programmes qui stockent la page graphique II (le dessin) et l'opération inverse.

```
8000 REM

STOCKER PAGE II

8005 HOME
8010 INPUT "NOM DE L'IMAGE ";IM$
8020 D$ = CHR$ (4)
8030 PRINT D$"BSAVE";IM$;",A$4000,L$2000"
8999 RETURN
9000 REM

CHARGER SUR PAGE II

9005 HOME
9010 INPUT "NOM DE L'IMAGE? ";IM$
9015 HGR2 : HCOLOR= 3
9020 D$ = CHR$ (4)
9030 PRINT D$"BLOAD";IM$;",A$4000"
9060 GET A$
9999 TEXT : HOME : RETURN
```

Reste le calcul et tracé de l'Image

IMAGE

On commence par calculer les coordonnées tridimensionnelles **APPARENTES**, c'est à dire les coordonnées cartésiennes des points dans un référentiel lié à l'observateur.

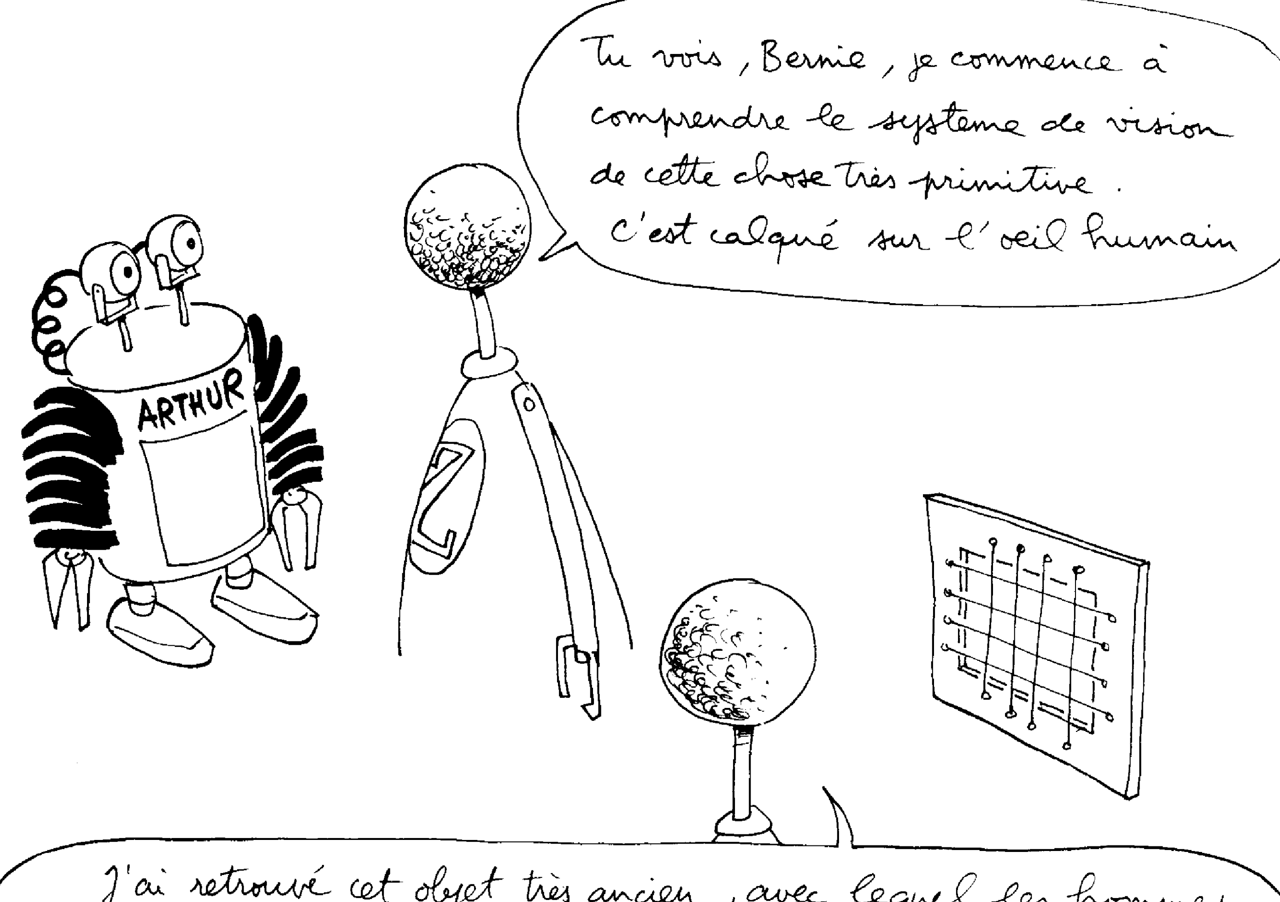
```
15000 REM

CALCUL ET TRACE IMAGE

15010 FOR I = 1 TO L: FOR J = 1 TO N(I)
15040 XL = XT(I,J) - XM
15050 YL = YT(I,J) - YM
15060 ZL = ZT(I,J) - ZM
15090 XA = XL * XU + YL * YU + ZL * ZU
15100 YA = XL * XV + YL * YV + ZL * ZV
15110 ZA = XL * XW + YL * YW + ZL * ZW
```

Comment convertir ces coordonnées tridimensionnelles en coordonnées de points sur l'écran.
On peut utiliser deux systèmes de repérage de visée.
le système de l'artilleur : azimut + site.
ou la visée **POLAIRE**.

le repérage de l'artilleur est **BESTIAL**. Il suffit de se reporter aux pages 12 et suivantes



— Mais j'ai trouvé aussi cet objectif **FISH-EYE** : oeil de poisson.



Son **OUVERTURE** atteint 90°.

— Tu vois, j'ai l'impression que les hommes ont fait, dès le départ, une erreur fondamentale sur le mécanisme de la vision. La **VISION** est un **GESTE**. En effet un oeil humain immobilisé est pratiquement aveugle. Pour plusieurs raisons. La première est que les cellules sensibles de la rétine sont principalement massées au voisinage de la tache fovéale, de l'axe optique. Et la seconde que ces cellules réagissent non à l'intensité lumineuse, mais à la **VARIATION** de cette intensité. La perception visuelle n'est donc pas statique. L'axe optique humain est en perpétuelle agitation.

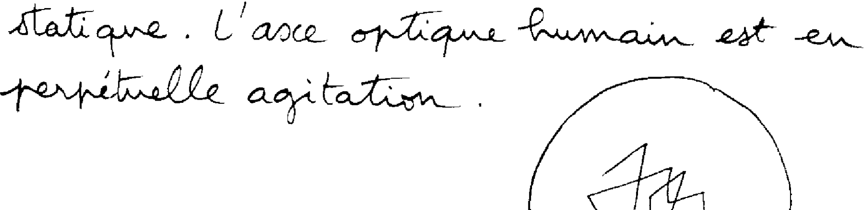
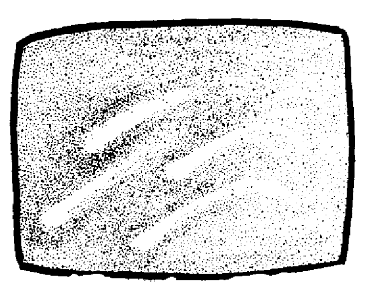


SCHÉMA DE LA RÉTINE

mouvement de l'axe optique.

— En somme, on voit mal le lien entre ce que les hommes montrent par exemple dans leurs films, et les images peuplant leur rétine ....

Un écran de Télévision a une certaine rémanence. Lorsque la caméra filme un ensemble de lampes donnant un éclairage très vif, et que cette caméra bouge, ces points lumineux laissent des traînées visibles sur l'écran fluorescent.

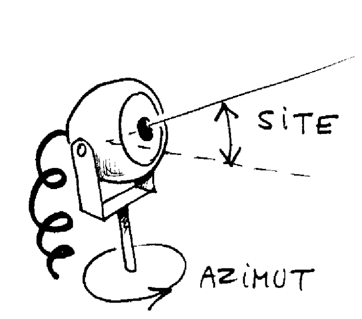


TRACE DU MOUVEMENT DE LAMPES SUR UN ÉCRAN, DÛ À LA RÉMANENCE

– En somme, l'image ci-contre donne une certaine idée de l'information dont dispose le cerveau humain pour reconstituer son environnement visuel.

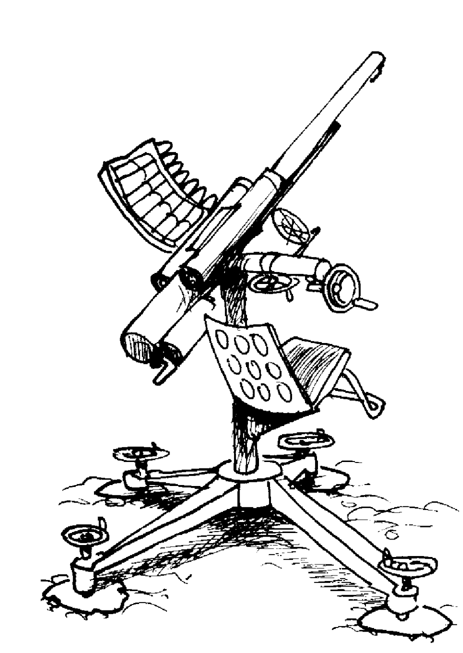
– La **VISION** est une **EXPLORATION** permanente du **CHAMP VISUEL**. L'axe optique est sans cesse dirigé vers des points d'intérêt.

– L'oeil d'Arthur est équipé pour réaliser cette sorte de **POURSUITE** de l'objet.



– Mais je ne suis pas sûr que ces angles : **AZIMUT, SITE**, traduisent la réalité mentale qui était celle des humains. Le pointage évoque plus le canon que l'oeil.

– Pourtant les hommes se basent sur ces déplacements en azimut et site, pour déplacer ces canons. Regarde cet antique système de DCA.

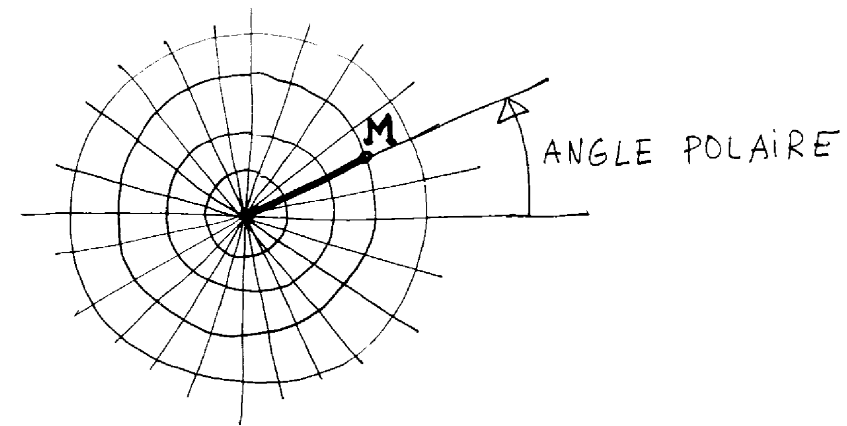


– Oui, mais regarde le **VISEUR** !

– Ah, tu as raison. Il matérialise une **POURSUITE** en **COORDONNÉES POLAIRES**.

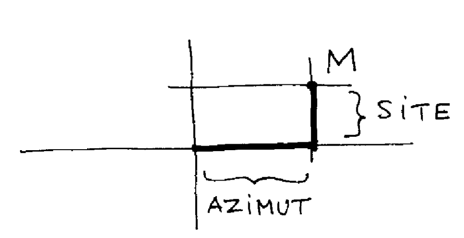
– Il n'y a pas de viseur à mailles carrées ou rectangulaires.

— Ainsi les hommes on construit des systèmes de **POINTAGE** basés sur le système (azimut, site) pour des raisons de commodités mécaniques. Mais, quand il s'agit de **VISÉE**, ils retrouvent le système qui leur est le plus **NATUREL**, c'est à dire un système de coordonnées **POLAIRES**.



Dans l'opération **DESSIN**, l'écart angulaire vis à vis de l'axe visuel est converti en longueur ρ.

De même, précédemment, l'azimut et le site étaient convertis en **COORDONNÉES ÉCRAN X, Y**



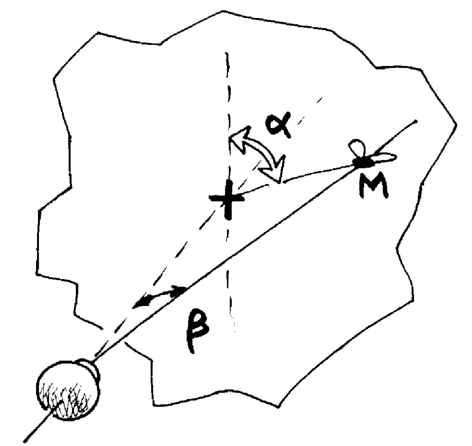
— Tu sais quoi ? On devrait modifier le programme de construction de l'image, en la définissant sur la base de ces **COORDONNÉES POLAIRES**.

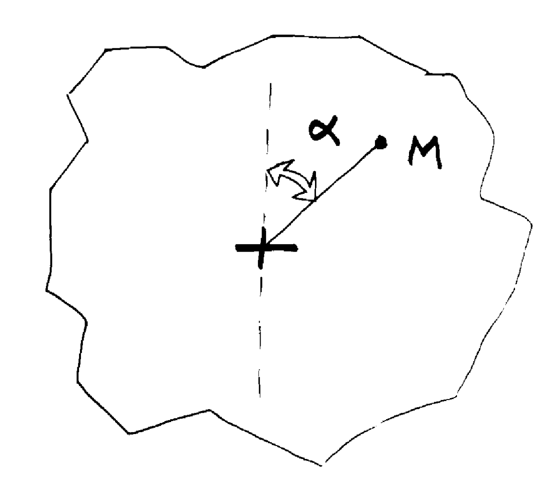
— Oui, cela serait plus naturel ....

— Et cela permet d'introduire des **OUVERTURES ANGULAIRES** plus importantes.

— Dans cette **FENÊTRE DE VISÉE**, moi je fais de la claustrophobie ...

VISÉE POLAIRE

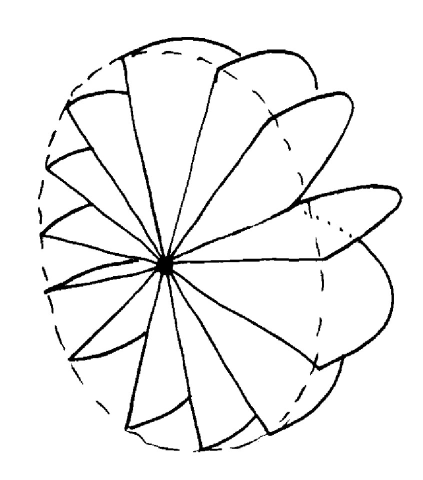


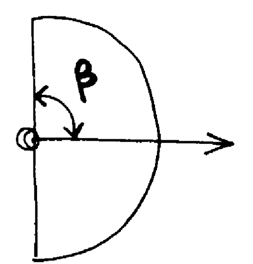


La croix est la trace de l'axe visuel.
α est l'angle polaire
β est l'écart angulaire, vis à vis de la direction axial

Un tel système permet de rendre compte de visées dépassant les capacités des yeux humains.

LE FISH-EYE

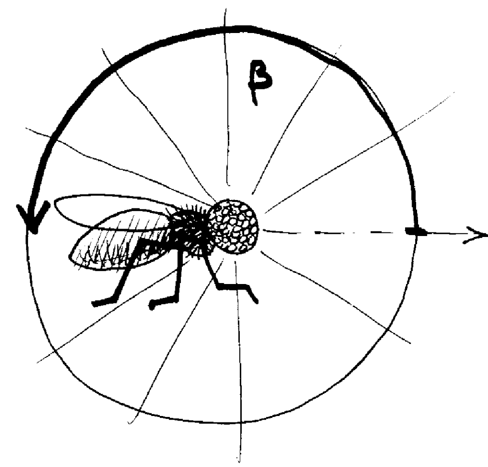




L'écart angulaire peut alors atteindre 90°

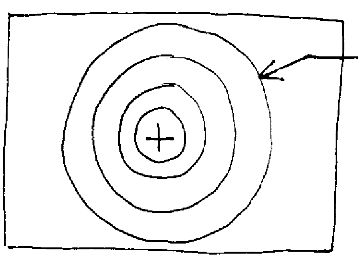
Mais on peut pousser plus loin encore.

L'ŒIL DE MOUCHE

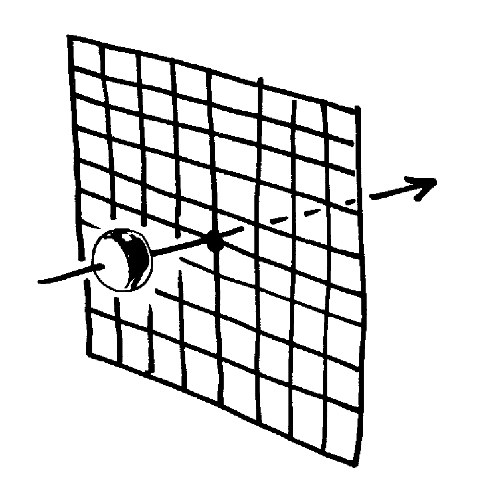


La déviation angulaire β peut alors atteindre la valeur maximale 180°

Dessiner sera donner une représentation plane, en coordonnées polaires de ces images. Le champ visuel est alors représenté selon un cercle. Et le rayon de ce cercle correspond à la valeur maximale de l'écart angulaire β. Sur l'écran du moniteur nous aurons :

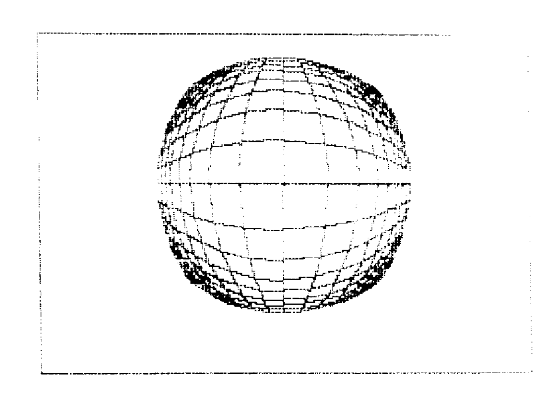


Dans le cas du FISH-EYE le cercle limite correspond à $\beta = 90°$.

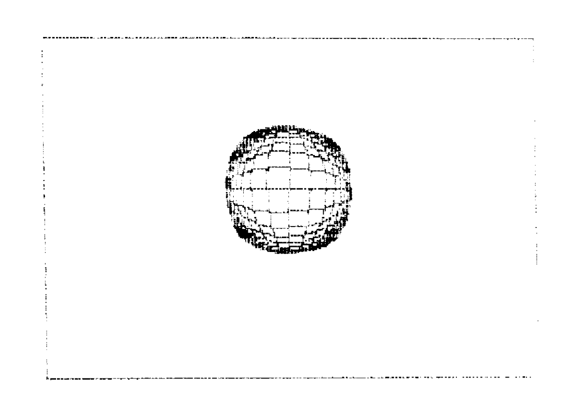


Un observateur équipé d'un FISH-EYE observe une mire carrée, maillée, dont le plan est perpendiculaire à son axe visuel.

La représentation plane est donnée sur la page suivante.

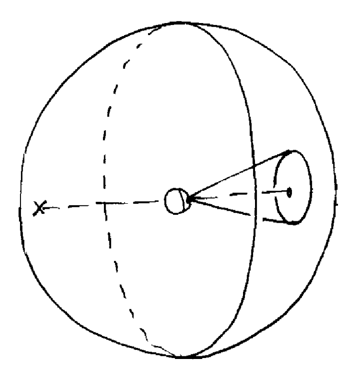


Passons maintenant en vision totale (oeil de la mouche.)
La représentation plane de cet objet serait :



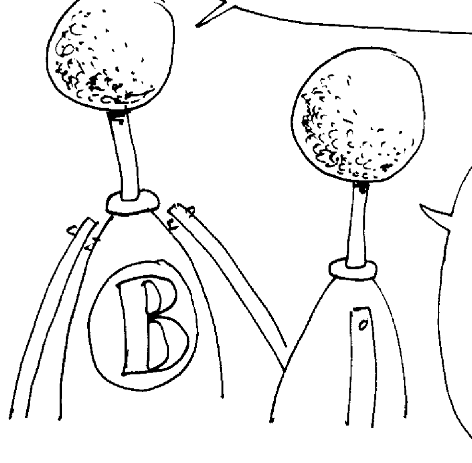
Il s'est tassé au centre de l'écran, puisque les valeurs de β de ses points sont relativement faibles vis à vis de la valeur maximale $\beta = 180°$

Nous découvrons ici un aspect déconcertant de cette représentation. Le point qui est situé "derrière la tête de l'observateur" se matérialise selon un cercle ($\beta = 180°$)
Convenons d'appeler le point a l'infini dans l'axe de visée le **POINT FRONTAL**, et le point situé dans la direction diamètralement le **POINT OCCIPITAL.**
L'image plane du point occipital est donc un cercle.



Si on limite l'écart angulaire maximum à une vingtaine ou à une trentaine de degrés on retrouve une vision très proche de la vision humaine.

On va donc réécrire le programme **PANGRAPHE** de telle manière qu'il puisse intégrer tous ces types de vision, de celle de l'homme à celle de la mouche, en passant par le fish-eye.





– l'Apple II trace les segments à l'aide de l'instruction

```
HPLOT XA,YA TO XB,YB
```

– c'est à dire qu'il opère sur des coordonnées-écran cartésiennes, qu'il faudra calculer. Mais, dans un mode de tracé **VECTORIEL**, on pourrait utiliser directement des données polaires.

– Quand les humains peuplaient encore la Terre, certains savaient dessiner, et d'autres en étaient parfaitement incapables. Pourquoi ?

– A mon avis, il devait s'agir d'un mauvais accord entre leur hard et leur soft. Le dessin manuel suggère une exploitation de données en coordonnées polaires. Le dessin est un geste. La main se déplace dans la direction polaire, et l'amplitude du geste traduit l'écart angulaire.

– Autrement dit, si un type code dans sa tête en coordonnées cartésiennes, il lui est impossible de dessiner...

– C'est un problème d'interfaçage.

– Sans doute.



la séquence suivante (qui lève aussi les ambiguïtés sur les Arctangentes) matérialise ce calcul, cette projection sur la sphère d'observation-écran, par visée-polaire.

```
15121  IF XA = 0 AND YA = 0 AND ZA = 0 THEN ZA = .001
15122 RO =  SQR (YA * YA + ZA * ZA)
15125  IF XA = 0 THEN B = 90: GOTO 15127
15126 B =  ATN (RO / XA):B = B * 180 / 3.1416
15127  IF XA < 0 THEN B = B + 180
15130  IF YA = 0 AND ZA = 0 THEN A = 90: GOTO 15180
15140  IF ZA = 0 AND YA > 0 THEN A =  - 90: GOTO 15180
15150  IF ZA = 0 AND YA < 0 THEN A = 90: GOTO 15180
15160 A =  -  ATN (YA / ZA):A = A * 180 / 3.1416
15170  IF ZA < 0 AND YA > 0 THEN A = A - 180
15172  IF ZA < 0 AND YA < 0 THEN A = A + 180
15174  IF ZA < 0 AND YA = 0 THEN A = 180
15180 X = B *  SIN (A * 3.1416 / 180):Y = B *  COS (A * 3.1416
 / 180)
15240 X(I,J) = 140 + X * 85 / AN
15250 Y(I,J) = 85 - 85 * Y / AN
15260  NEXT J: NEXT I
```

X(I,J) et Y(I,J) sont les coordonnées-écran.

On va ensuite matérialiser les segments "AB" successifs constituant la chaîne à dessiner. Ils devront être analysés un à un (problème du FENÊTRAGE)

```
15280  FOR I = 1 TO L
15290  FOR J = 1 TO N(I) - 1
15300 XA = X(I,J):YA = Y(I,J):FP = 0:H = 0
15310 XB = X(I,J + 1):YB = Y(I,J + 1)
```

la séquence ci après représente le travail de fenêtrage. On serait en effet tenté de tracer le segment simplement en faisant

HPLOT XA,YA TO XB,YB

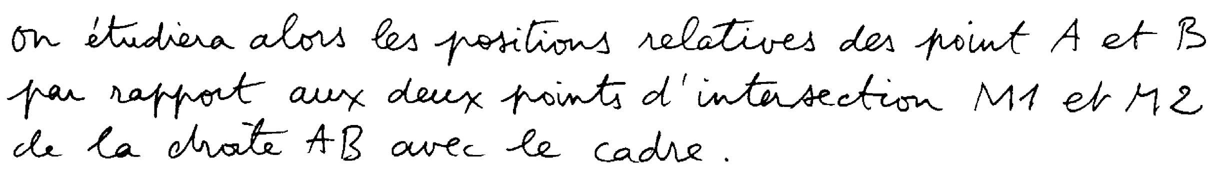
message qui ne sera exécuté que si les points A et B sont tous les deux à l'intérieur du cadre. Dans le cas contraire on récupèrera un message d'erreur.

FENÈTRAGE

```
15320 IF YA = YB AND YA >  = 0 AND YA <  = 190 THEN XL(1
) = 0:XL(2) = 27
   9:YL(1) = YA:YL(2) = YA:FP = 1: GOTO 15480
15325 IF XA = XB AND XA < 0 THEN 15590
15326 IF XA = XB AND XA > 279 THEN 15590
15330 IF XA = XB AND XA >  = 0 AND XA <  = 279 THEN XL(1)
= XA:XL(2) = X
   A:YL(1) = 0:YL(2) = 190:FP = 1: GOTO 15480
15335 IF YB = YA AND YA < 0 THEN 15590
15336 IF YB = YA AND YA > 190 THEN 15590
15340 P = (YB - YA) / (XB - XA)
15360 YI = YA - P * XA
15370 IF YI >  = 0 AND YI <  = 190 THEN FP = 1:H = H + 1:XL(
H) = 0:YL(H)
    = YI
15390 YI = YA + (279 - XA) * P
15400 IF YI >  = 0 AND YI <  = 190 THEN FP = 1:H = H + 1:XL(
H) = 279:YL(
   H) = YI
15420 XI = XA - YA / P
15430 IF XI >  = 0 AND XI <  = 279 THEN FP = 1:H = H + 1:XL(H
) = XI:YL(H
   ) = 0
15450 XI = XA + (190 - YA) / P
15460 IF XI >  = 0 AND XI <  = 279 THEN FP = 1:H = H + 1:XL(H
) = XI:YL(H
   ) = 190
15470 IF FP = 0 THEN 15590
15480 SA = (XA - XL(1)) * (XA - XL(2)) + (YA - YL(1)) * (YA - YL
(2))
15490 SB = (XB - XL(1)) * (XB - XL(2)) + (YB - YL(1)) * (YB - YL
(2))
15492 S1 = (XL(1) - XA) * (XL(1) - XB) + (YL(1) - YA) * (YL(1) - Y
B)
15494 S2 = (XL(2) - XA) * (XL(2) - XB) + (YL(2) - YA) * (YL(2) - Y
B)
15500 IF SA <  = 0 AND SB <  = 0 THEN  HPLOT XA,YA TO XB
,YB: GOTO 15590
15510 IF SA > 0 AND SB > 0 AND S1 < 0 AND S2 < 0 THEN  HP
LOT XL(1),YL(1)
    TO XL(2),YL(2): GOTO 15590
15520 IF SA <  = 0 THEN XD = XA:YD = YA
15530 IF SB <  = 0 THEN XD = XB:YD = YB
15560 IF S1 <  = 0 THEN XF = XL(1):YF = YL(1)
15570 IF S2 <  = 0 THEN XF = XL(2):YF = YL(2)
15580 HPLOT XD,YD TO XF,YF
15590 NEXT J: NEXT I
15999 RETURN
```

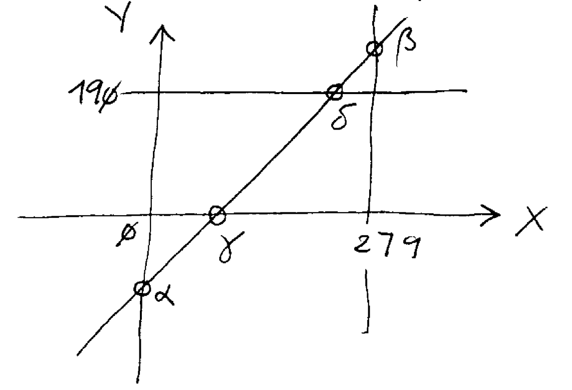
Dans les ordinateurs actuels il n'est plus nécessaire de programmer le fenêtrage : il est automatique.

... vestige du passé ...



On étudiera alors les positions relatives des point A et B par rapport aux deux points d'intersection M1 et M2 de la droite AB avec le cadre.

Redressons l'axe OY de l'écran (sur l'Apple II il a "la tête en bas").



La pente de la droite AB est P = (YB − YA)/(XB − XA)

l'équation de la droite AB est : $\dfrac{Y - YA}{X - XA} = \dfrac{YB - YA}{XB - XA} = P$

Premier élément du cadre $X = 0 \Rightarrow YI = YA - P * XA$ (α)

La droite AB ne peut couper le cadre qu'en deux points au plus. Ce point sera à retenir si : $0 < YI < 190$

Second element du cadre : $X = 279 \Rightarrow YI = YA + (279 - XA) * P$ (β)

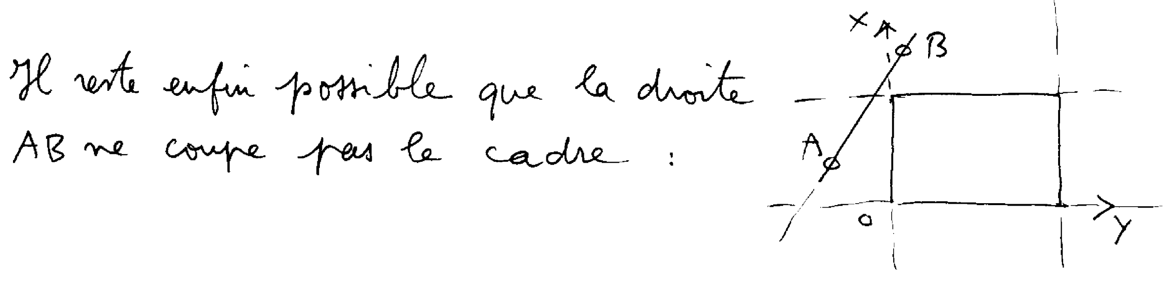
à retenir si $0 < YI < 190$

Troisième élement : $Y = 0 \Rightarrow XI = XA - YA / P$ (γ)

à retenir si $0 < XI < 279$

Quatrième élement $Y = 190 \Rightarrow XI = XA + (190 - YA) / P$ (δ)

à retenir si $0 < XI < 279$.

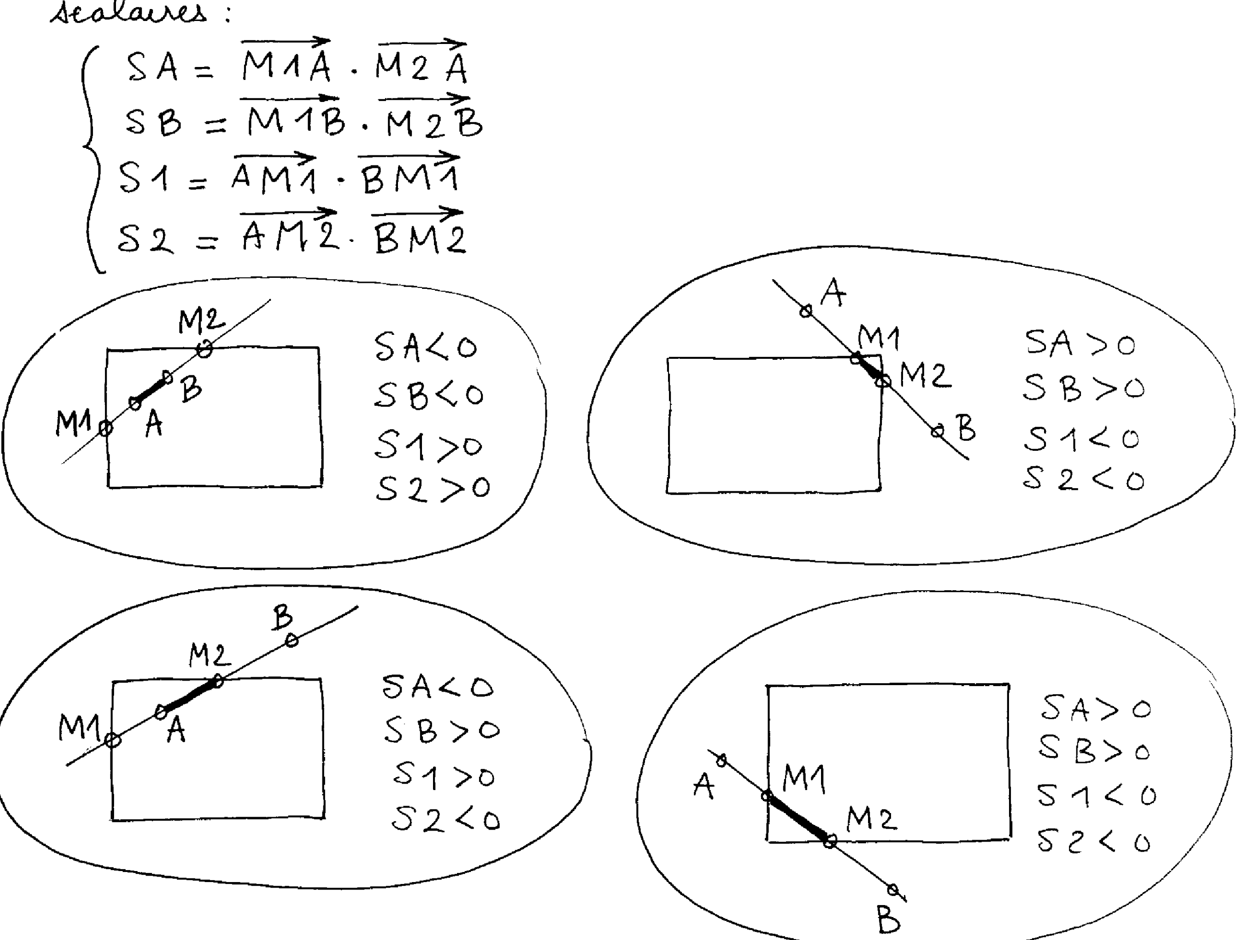


Il reste enfin possible que la droite AB ne coupe pas le cadre :

Comme on le verra plus loin, une petite astuce de programmation permet de déterminer en une seule fois les éventuels points d'intersection M1 et M2 de la droite AB avec le cadre. Il restera à déterminer les positions respectives des points A, B, M1, M2 sur la droite.

On se basera pour se faire sur le signe des produits scalaires :

$$\begin{cases} SA = \overrightarrow{M1A} \cdot \overrightarrow{M2A} \\ SB = \overrightarrow{M1B} \cdot \overrightarrow{M2B} \\ S1 = \overrightarrow{AM1} \cdot \overrightarrow{BM1} \\ S2 = \overrightarrow{AM2} \cdot \overrightarrow{BM2} \end{cases}$$



On voit comment l'examen des produits scalaires permet de choisir les extrémités du segment à tracer.

15315

FENETRAGE

M1 M2

```
15320 IF YA=YB AND YA>=Ø AND YA<=19Ø THEN
XL(1)=Ø : XL(2)=279 : YL(1)=YA : YL(2)=YA :
FP=1 : GOTO 1548Ø
```

```
15325 IF XA=XB AND XA<Ø THEN 15590
```

```
15326 IF XA=XB AND XA>279 THEN 1559Ø
```

M2 M1

```
1533Ø IF XA=XB AND XA>=Ø AND YA<=279
THEN XL(1)=XA : XL(2)=XA : YL(1)=Ø : YL(2)
=19Ø : FP=1 : GOTO 1548Ø
```

```
15335 IF YB=YA AND YA<Ø THEN 1559Ø
```

```
15336 IF YB=YA AND YA>19Ø THEN 1559Ø
```

Calcul de la pente

```
1534Ø P=(YB-YA)/(XB-XA)
```

Intersection avec la droite X=Ø

```
1536Ø YI = YA-P*XA
```

Y M1 X

le bord gauche est coupé par la droite AB ?

```
1537Ø IF YI>=Ø AND YI<=19Ø THEN FP=1:
H=H+1 : XL(H)=Ø : YL(H)=YI
```

Intersection avec la droite X=279

```
1539Ø YI=YA+(279-XA)*P
```

Y X

Test pour voir si le bord droit est coupé par AB

```
1540Ø IF YI>=Ø AND YI<=19Ø THEN FP=1:
H=H+1 : XL(H)=279 : YL(H)=YI
```

Intersection avec la droite Y=Ø

```
15420 XI = XA - YA/P
```

Test d'intersection avec le bord situé dans Y=Ø

```
15430 IF XI >= Ø AND XI <= 279 THEN FP=1:
H=H+1 : XL(H) = XI : YL(H)=Ø
```

Intersection avec la droite Y=19Ø

```
15450 XI = XA + (19Ø - YA) /P
```

Test

```
15460 IF XI >=Ø AND XI<= 279 THEN FP=1:
H=H+1 : XL(H) = XI : YL(H)=19Ø
```

Cas où AB ne coupe pas le cadre :

(Flag resté nul)

```
15470 IF FP=Ø THEN 1559Ø
```

Produit Scalaire $\overrightarrow{M1A} \cdot \overrightarrow{M2A}$

```
15480 SA = (XA - XL(1)) * (XA - XL(2))
+ (YA - YL(1))*(YA-YL(2))
```

Produit Scalaire $\overrightarrow{M1B} \cdot \overrightarrow{M2B}$

```
15490 SB = (XB - XL(1)) * ( XB - XL(2))
+ (YB - YL(1)) * (YB - YL(2))
```

Produit Scalaire $\overrightarrow{AM1} \cdot \overrightarrow{BM1}$

```
15492 S1 = (XL(1) - XA)*(XL(1) - XB)
+ (YL(1) - YA) * ( YL(1) - YB)
```

Produit Scalaire $\overrightarrow{AM2} \cdot \overrightarrow{BM2}$

```
15494 S2 = ( XL(2)-XA) ( XL(2) - XB)
+(YL(2) - YA) *( YL(2) - YB)
```

A et B à l'intérieur du segment M1M2

```
155ØØ IF SA <= Ø AND SB <= Ø THEN
HPLOT XA,YA TO XB,YB : GOTO 15590
```

M1 M2

A B

M1 et M2 sont à l'intérieur de AB

```
15510 IF SA>0 AND SB>0 AND S1<0
AND S2<0 THEN HPLOT XL(1), YL(1) TO
XL(2), YL(2) : GOTO 15590
```

B

A

M1 M2

```
15520 IF SA<=0 THEN XD=XA : YD=YA
```

(début du segment à tracer)

```
15530 IF SB<=0 THEN XD=XB : YD=YB
15540 IF S1<=0 THEN XF=XL(1) : YF=YL(1)
15550 IF S2<=0 THEN XF=XL(2) : YF=YL(2)
15580 HPLOT XD, YD TO XF, YF
15590 NEXT J : NEXT I
15999 RETURN
```

Bon, maintenant nous avons déja des éléments pour commencer à dessiner

Crois-tu, Zaks, que nous allons avoir des émotions artistiques ?

Je ne sais pas... on verra bien...

PANGRAPHE DESSIN

```
0  IF FD = 1 THEN 5
1 L = 0: REM PANGRAPHE DESSIN 5/2/84
2  DIM XT(30,9),YT(30,9),ZT(30,9),X(30,9),Y(30,9),N(30):FD = 1
3 D$ =  CHR$ (13) +  CHR$ (4)
5  TEXT : HOME
6  IF C <  > 0 THEN 330
10  HOME
20  REM

 PROGRAMME MENU


30  VTAB 15: HTAB 15
40  PRINT "VOULEZ-VOUS:"
50  FOR TE = 1 TO 500: NEXT
60  HOME
70  PRINT
80  VTAB 3: HTAB 25: PRINT "PANGRAPHE DESSIN"
90  VTAB 1
100  PRINT "1-(CREER UN OBJET)"
110  PRINT "2-(STOCKER UN OBJET)"
120  PRINT "3-CHARGER UN OBJET"
130  PRINT "4-(COMPLETER UN OBJET)"
140  PRINT "5-(MODIFIER UN OBJET)
150  PRINT "6-REPRESENTER UN OBJET"
160  PRINT "7-STOCKER UNE IMAGE"
170  PRINT "8-CHARGER UNE IMAGE"
180  PRINT "9-(MANIPULER UN OBJET)"
190  PRINT "10-(DEFINIR UN BLOC D'OBJETS)"
200  PRINT "11-CONSULTER UN FICHIER BLOC"
210  PRINT "12-"
220  PRINT "13-"
230  PRINT "14-"
240  PRINT "15-"
250  PRINT "16-"
260  PRINT "17-AFFICHER LE CATALOGUE"
270  PRINT "18-SORTIR DU PROGRAMME"
290  INPUT "VOTRE CHOIX ";C
300  IF C = 17 THEN  PRINT D$"CATALOG": GET A$: GET A$
310  IF C = 18 THEN  END
320  IF C = 19 THEN  PRINT D$"RUNPANGRAPHE PRINCIPAL
"
330  ON C GOSUB 40000,40000,4000,40000,40000,7000,8000,90
00,40000,40000,2
   9000
999  GOTO 10
4000  REM

CHARGEMENT OBJET


4010  INPUT "NOM DE L'OBJET ";O$
4020 D$ =  CHR$ (4)
4030  PRINT D$"OPEN";O$
4040  PRINT D$"READ";O$
4050  INPUT L
4060  FOR I = 1 TO L
4070  INPUT N(I)
4080  NEXT I
4090  FOR I = 1 TO L
4100  FOR J = 1 TO N(I)
4110  INPUT XT(I,J): INPUT YT(I,J): INPUT ZT(I,J)
4120  NEXT J: NEXT I
4130  PRINT D$"CLOSE";O$
4999  RETURN
5000  GOSUB 2000
5999  RETURN
```

```
7000 REM

REPRESENTER UN OBJET


7010 TEXT : HOME
7020 PRINT : PRINT
7030 INPUT "OUVERTURE ANGULAIRE ";AN: PRINT
7040 PRINT "POINT D'OBSERVATION"
7050 PRINT : INPUT "X=";XM
7060 INPUT "Y=";YM
7070 INPUT "Z=";ZM
7080 PRINT : PRINT "CENTRE G DE L'IMAGE": PRINT : INPU
T "XG=";XG: INPUT
   "YG=";YG: INPUT "ZG=";ZG
7090 CX = XG - XM:CY = YG - YM:CZ = ZG - ZM
7100 DI = SQR (CX * CX + CY * CY)
7110 IF DI = 0 AND CZ > 0 THEN TE = 0:KI = 3.1416 / 2: GOTO
 7210
7120 IF DI = 0 AND CZ < 0 THEN TE = 0:KI = - 3.1416 / 2: GO
TO 7210
7130 KI = ATN (CZ / DI)
7140 IF CX = 0 AND CY > 0 THEN TE = 1.57: GOTO 7210
7150 IF CX = 0 AND CY < 0 THEN TE = - 1.57: GOTO 7210
7160 TE = ATN (CY / CX)
7170 IF CX < 0 AND CY > 0 THEN TE = 3.1416 + TE
7180 IF CX < 0 AND CY < 0 THEN TE = 3.1416 + TE
7190 IF CX < 0 AND CY = 0 THEN TE = 3.1416
7200 IF CX = 0 AND CY < 0 THEN TE = - 3.1416 / 2
7210 AZ = TE * 170 / 3.1416:SI = KI * 170 / 3.1416
7220 REM AZ AZIMUT;SI SITE
7230 CT = COS (TE):ST = SIN (TE)
7240 CK = COS (KI):SK = SIN (KI)
7250 XU = CK * CT:YU = CK * ST:ZU = SK
7260 XV = - ST:YV = CT:ZV = 0
7270 XW = - SK * CT:YW = - SK * ST:ZW = CK
7280 HOME
7290 PRINT "1-OBJET DEJA EN MEMOIRE": PRINT
7300 PRINT "2-DESSIN PAR ELEMENTS": PRINT
7310 PRINT "3-DESSIN PAS A PAS": PRINT
7320 PRINT "4-DESSIN PAR BLOC": PRINT
7330 INPUT "VOTRE CHOIX ";C: PRINT
7340 ON C GOSUB 12000,11000,30000,10000
7998 GET A$: TEXT
7999 RETURN
8000 REM

STOCKER PAGE II


8005 HOME
8010 INPUT "NOM DE L'IMAGE ";IM$
8020 D$ = CHR$ (4)
8030 PRINT D$"BSAVE";IM$;",A$4000,L$2000"
8999 RETURN
9000 REM

CHARGER SUR PAGE II


9005 HOME
9010 INPUT "NOM DE L'IMAGE? ";IM$
9015 HGR2 : HCOLOR= 3
9020 D$ = CHR$ (4)
9030 PRINT D$"BLOAD";IM$;",A$4000"
9060 GET A$
9999 TEXT : HOME : RETURN
10000 REM

DESSIN PAR BLOC


10010 INPUT "NOM DU BLOC ";BL$: PRINT
10030 HGR2 : HCOLOR= 3
10040 HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
10050 FOR K = 1 TO E
10060 O$ = EL$(K)
10070 GOSUB 4020
10080 GOSUB 15000
10090 NEXT K
10100 FOR T = 1 TO 7: PRINT CHR$ (7): NEXT T
10999 RETURN
```

```
11000  REM

DESSIN PAR ELEMENTS


11010  HOME
11020  INPUT "NOMBRE D'ELEMENTS ";E
11030  FOR K = 1 TO E
11040  INPUT "NOM ELEMENT ";EL$(K)
11045  NEXT K
11050  HGR2 : HCOLOR= 3
11060  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
11070  FOR K = 1 TO E
11080 O$ = EL$(K)
11090  GOSUB 4020
11100  GOSUB 15000
11110  NEXT K
11120  FOR T = 1 TO 7: PRINT  CHR$ (7): NEXT T
11999  RETURN
12000  REM

OBJET DEJA EN MEMOIRE


12010  HGR2 : HCOLOR= 3
12020  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
12030  GOSUB 15000
12040  FOR T = 1 TO 7: PRINT  CHR$ (7): NEXT T
12999  RETURN
15000  REM

CALCUL ET TRACE IMAGE


15010  FOR I = 1 TO L: FOR J = 1 TO N(I)
15040 XL = XT(I,J) - XM
15050 YL = YT(I,J) - YM
15060 ZL = ZT(I,J) - ZM
15090 XA = XL * XU + YL * YU + ZL * ZU
15100 YA = XL * XV + YL * YV + ZL * ZV
15110 ZA = XL * XW + YL * YW + ZL * ZW
15121  IF XA = 0 AND YA = 0 AND ZA = 0 THEN ZA = .001
15122 RO =  SQR (YA * YA + ZA * ZA)
15125  IF XA = 0 THEN B = 90: GOTO 15127
15126 B =  ATN (RO / XA):B = B * 180 / 3.1416
15127  IF XA < 0 THEN B = B + 180
15130  IF YA = 0 AND ZA = 0 THEN A = 90: GOTO 15180
15140  IF ZA = 0 AND YA > 0 THEN A =  - 90: GOTO 15180
15150  IF ZA = 0 AND YA < 0 THEN A = 90: GOTO 15180
15160 A =  -  ATN (YA / ZA):A = A * 180 / 3.1416
15170  IF ZA < 0 AND YA > 0 THEN A = A - 180
15172  IF ZA < 0 AND YA < 0 THEN A = A + 180
15174  IF ZA < 0 AND YA = 0 THEN A = 180
15180 X = B *  SIN (A * 3.1416 / 180):Y = B *  COS (A * 3.1416
 / 180)
15240 X(I,J) = 140 + X * 85 / AN
15250 Y(I,J) = 85 - 85 * Y / AN
15260  NEXT J: NEXT I
15280  FOR I = 1 TO L
15290  FOR J = 1 TO N(I) - 1
15300 XA = X(I,J):YA = Y(I,J):FP = 0:H = 0
15310 XB = X(I,J + 1):YB = Y(I,J + 1)
15320  IF YA = YB AND YA >  = 0 AND YA <  = 190 THEN XL(1
) = 0:XL(2) = 27
    9:YL(1) = YA:YL(2) = YA:FP = 1: GOTO 15480
15325  IF XA = XB AND XA < 0 THEN 15590
15326  IF XA = XB AND XA > 279 THEN 15590
15330  IF XA = XB AND XA >  = 0 AND XA <  = 279 THEN XL(1)
= XA:XL(2) = X
    A:YL(1) = 0:YL(2) = 190:FP = 1: GOTO 15480
15335  IF YB = YA AND YA < 0 THEN 15590
15336  IF YB = YA AND YA > 190 THEN 15590
15340 P = (YB - YA) / (XB - XA)
15360 YI = YA - P * XA
15370  IF YI >  = 0 AND YI <  = 190 THEN FP = 1:H = H + 1:XL(
H) = 0:YL(H)
     = YI
15390 YI = YA + (279 - XA) * P
15400  IF YI >  = 0 AND YI <  = 190 THEN FP = 1:H = H + 1:XL(
H) = 279:YL(
    H) = YI
15420 XI = XA - YA / P
15430  IF XI >  = 0 AND XI <  = 279 THEN FP = 1:H = H + 1:XL(H
) = XI:YL(H
    ) = 0
15450 XI = XA + (190 - YA) / P
15460  IF XI >  = 0 AND XI <  = 279 THEN FP = 1:H = H + 1:XL(H
) = XI:YL(H
    ) = 190
15470  IF FP = 0 THEN 15590
15480 SA = (XA - XL(1)) * (XA - XL(2)) + (YA - YL(1)) * (YA - YL
(2))
```

```
15490 SB = (XB - XL(1)) * (XB - XL(2)) + (YB - YL(1)) * (YB - YL
(2))
15492 S1 = (XL(1) - XA) * (XL(1) - XB) + (YL(1) - YA) * (YL(1) - Y
B)
15494 S2 = (XL(2) - XA) * (XL(2) - XB) + (YL(2) - YA) * (YL(2) - Y
B)
15500  IF SA < = 0 AND SB < = 0 THEN  HPLOT XA,YA TO XB
,YB: GOTO 15590
15510  IF SA > 0 AND SB > 0 AND S1 < 0 AND S2 < 0 THEN  HP
LOT XL(1),YL(1)
    TO XL(2),YL(2): GOTO 15590
15520  IF SA < = 0 THEN XD = XA:YD = YA
15530  IF SB < = 0 THEN XD = XB:YD = YB
15560  IF S1 < = 0 THEN XF = XL(1):YF = YL(1)
15570  IF S2 < = 0 THEN XF = XL(2):YF = YL(2)
15580  HPLOT XD,YD TO XF,YF
15590  NEXT J: NEXT I
15999  RETURN
29000  REM
LECTURE FICHIER BLOC


29005  HOME : INPUT "NOM DU BLOC ";BL$: PRINT
29010 D$ =  CHR$ (4)
29020  PRINT D$"OPEN";BL$
29030  PRINT D$"READ";BL$
29040  INPUT E
29050  FOR I = 1 TO E
29060  INPUT EL$(I)
29065  NEXT I
29070  PRINT D$"CLOSE";BL$
29080  FOR I = 1 TO E: PRINT EL$(I): NEXT I
29090  FOR TT = 1 TO 2000: NEXT TT
29999  RETURN
30000  REM

DESSIN PAS A PAS


30010  HOME
30020  INPUT "NOM PREMIER ELEMENT ";O$: PRINT
30030  GOSUB 4020
30040  HGR2 : HCOLOR= 3
30050  HPLOT 0,0 TO 279,0 TO 279,190 TO 0,190 TO 0,0
30060  GOSUB 15000: FOR TT = 1 TO 7: PRINT  CHR$ (7): NEXT
 TT
30070  GET A$: GET A$
30080  POKE  - 16303,0: POKE  - 16300,0: REM TEXTE
30090  INPUT "UN AUTRE ELEMENT ? ";R$: PRINT
30100  IF  LEFT$ (R$,1) < > "O" THEN 30999
30110  INPUT "NOM DE L'ELEMENT ";O$: PRINT
30120  GOSUB 4020
30130  POKE  - 16304,0: POKE  - 16299,0: REM PAGEII
30140  GOSUB 15000: FOR TT = 1 TO 7: PRINT  CHR$ (7): NEXT
 TT
30150  GET A$: GET A$
30160  GOTO 30080
30999  RETURN
40000  REM

CHAINAGE SUR P/OBJET
40010  PRINT D$"BLOADCHAIN,A520"
40020  CALL 520"PANGRAPHE OBJET"
```

Remarque :

l'ensemble **PANGRAPHE-OBJET + PANGRAPHE DESSIN** a été conçu pour pouvoir être aisément récupéré et modifié par l'utilisateur, qui trouvera là une initiation à la **CONCEPTION ASSISTEE SUR ORDINATEUR (CAO).** Les Menus comportent des "lignes vides", que le lecteur pourra remplir au gré de son imagination.

1999 : Le **BASIC** moderne ne comportent plus de "lignes programmes". Les têtes de sous-programmes sont repérées à l'aide "d'étiquettes", ce qui donne encore plus de souplesse à l'ensemble. Toujours est-il que l'ensemble proposé est **MODULAIRE** et que l'utilisateur peut ajouter des sous-programmes à l'infini pour enrichir celui-ci.

Si vous dotez **PANGRAPHE DESSIN** de quelques lignes supplémentaires, vous aurez vite des ennuis.
La mémoire de l'Apple II s'organise comme ceci :

Les ennuis vont se traduire par l'apparition de tas de choses bizarres sur la page II. La raison est simple. Le programme BASIC, après avoir envahi la page I (non utilisée pour le graphisme) se met à grignoter votre page 2.
Une solution consiste à mettre en tête de programme

```
Ø LOMEM : 24576
```

Ceci a pour effet de tout reporter, programme, fichiers, tout ce que voulez faire ingurgiter à la machine, au dessus de la page II :

C'est une bonne solution.
Mais vous ne pourrez plus, en utilisant **CHAIN**, conserver les fichiers

PANGRAPHE DESSIN est en fait bien loin d'être achevé, puisqu'il manque tout le traitement lignes vues, lignes cachées, ombrages, etc...

Suggestion : créer un troisième programme **PANGRAPHE DESSIN II** qui pourra être appelée par la première partie de PANGRAPHE DESSIN.

On peut ainsi, si on dispose de deux drives, manoeuvrer in fine un outil dont l'énorme programme peut utiliser une part importante d'une diskette, tandis que l'objet, attaqué à travers le drive II pourra être constitué de milliers de segments.

au point de vue saisie vous pourrez gagner beaucoup de temps si vous disposez d'un digitaliseur (table graphique).

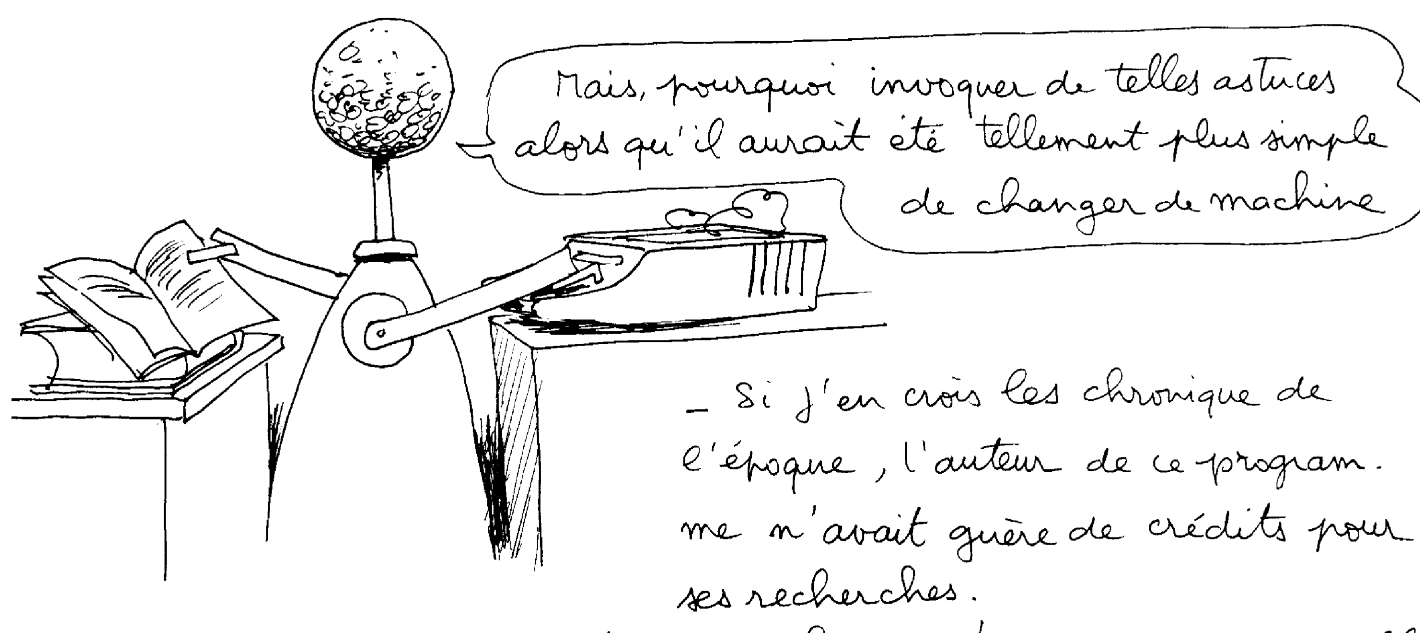
Avec une **TABLE TRAÇANTE** vous pourrez travailler très finement, au dixième de millimètre. Mais avec une simple imprimante faisant de la **RECOPIE D'ÉCRAN** vous pourrez conserver des copies papier de vos dessins.

A ce sujet, une astuce possible = au lieu de pointer vers le point G vous pouvez effectuer quatre dessins, au prix de quatre visées successives en

G_1 ×	G_2 ×
G_3 ×	G_4 ×

(G au centre)

G_1 G_2 G_3 G_4. Tirez les images sur papier, et assemblez. Vous quadruplez votre pouvoir de resolution. Et c'est comme si vous travailliez avec un Apple ayant une matrice graphique de 560 X 380 points...

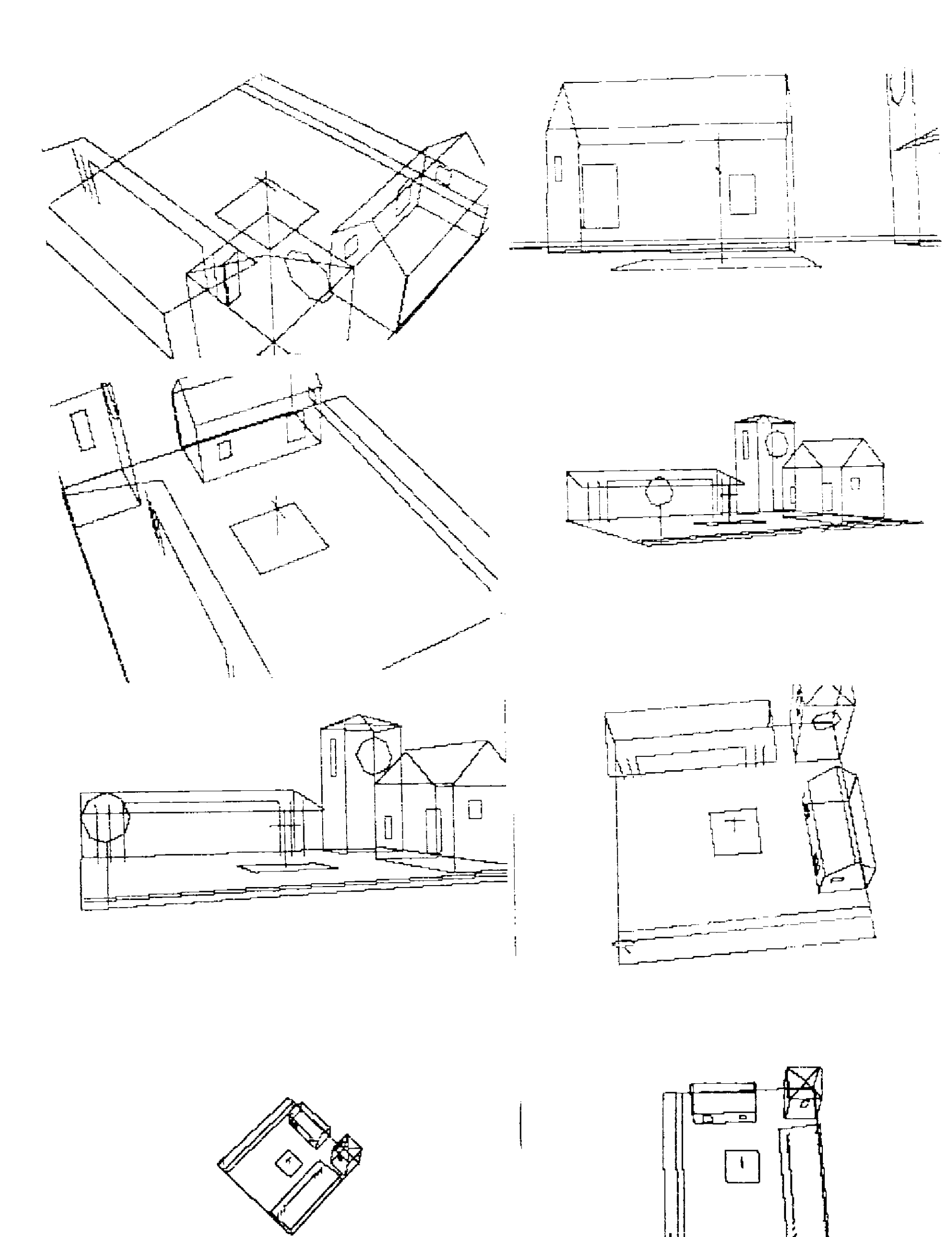


– Si j'en crois les chronique de l'époque, l'auteur de ce programme n'avait guère de crédits pour ses recherches.

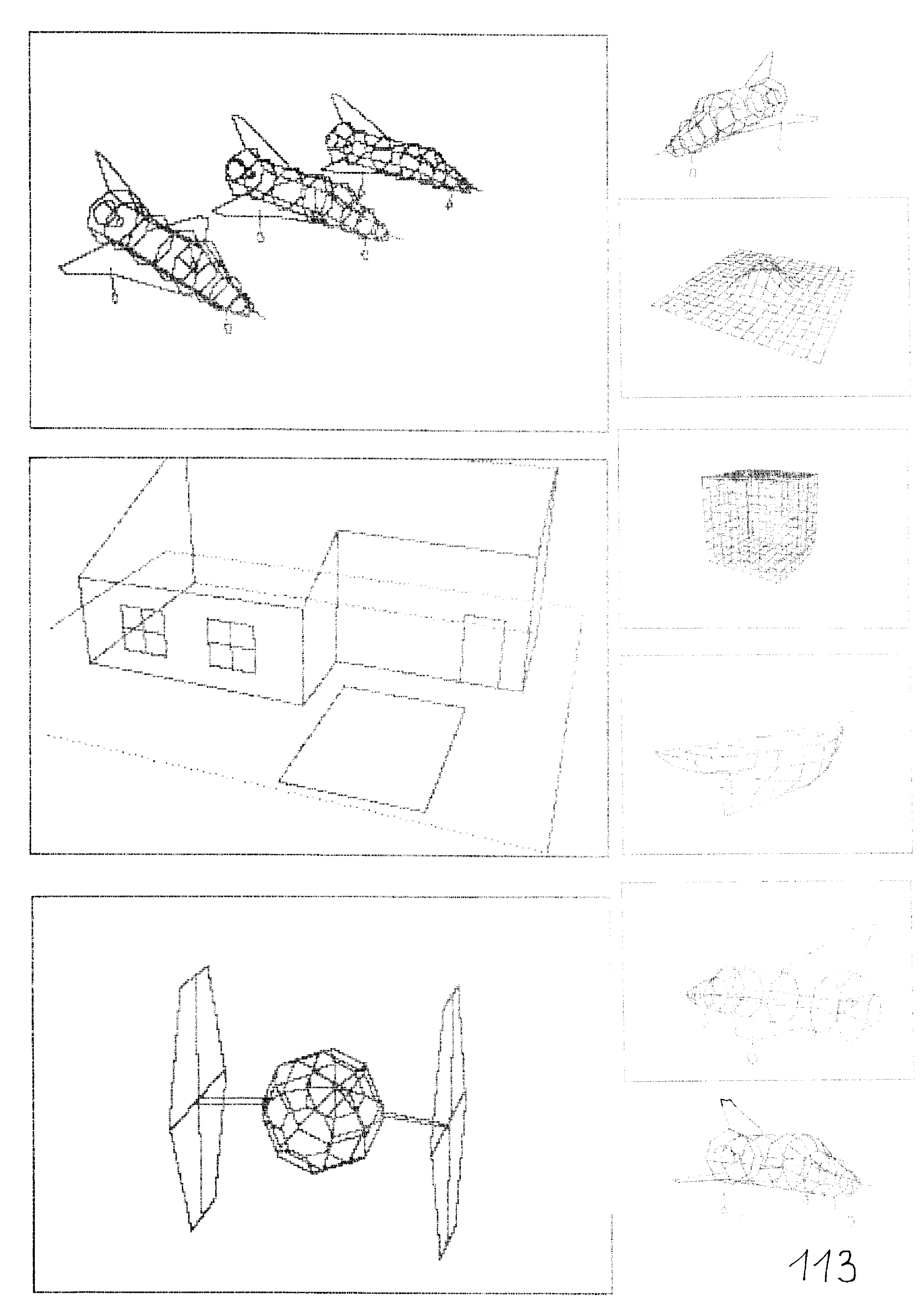
– Ça devait être en tout cas un homme très soigneux, capable de rentrer toutes ces lignes programme bout à bout sans se tromper !..

– C'est assez étonnant, quand on voit la photo de son laboratoire ...

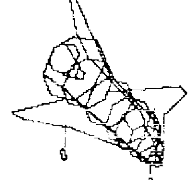
– Peut-être qu'il se trompait souvent, mais que c'était un homme d'un grand courage ...

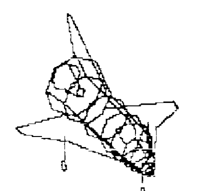


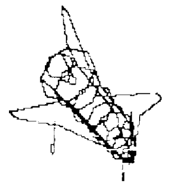
Une place de village, avec son clocher, son marché, sa mairie, créée par **PANGRAPHE**. Différentes vues extraites d'une animation.



SUITES D'IMAGES
CRÉÉES PAR
PANGRAPHE
POUR UNE
ANIMATION







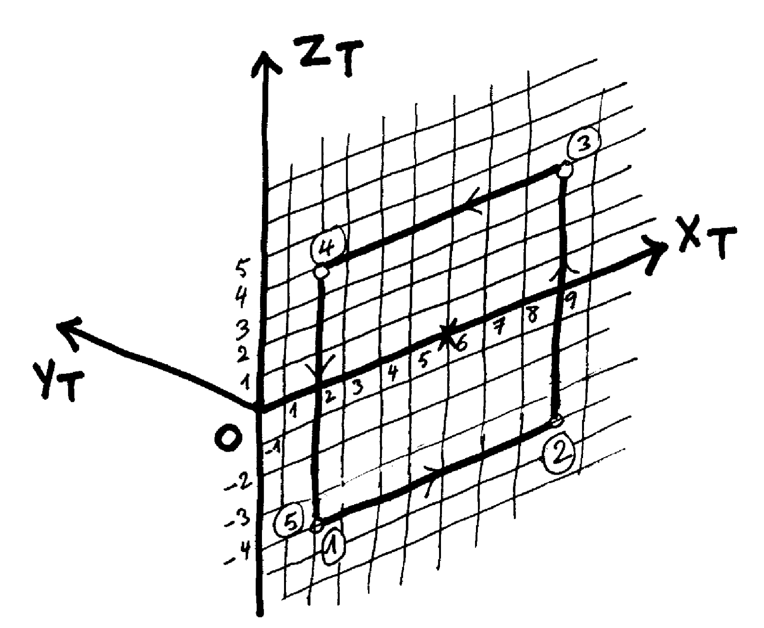
COMMENT UTILISER PANGRAPHE

La diskette porte deux programmes :

PANGRAPHE OBJET
PANGRAPHE DESSIN

Vous pouvez lancer au départ l'un ou l'autre de ces programmes puisque le **CHAIN** et le sous-programme de transfert 40.000 permettra de travailler comme si c'était un programme unique. Nous allons commencer par **PANGRAPHE OBJET.**

On fait donc RUN PANGRAPHE OBJET
Puis on choisit l'option 1-CREER UN OBJET
et la sous-option 1-CREER DES CHAINES
Nous allons maintenant créer un carré situé comme ceci :



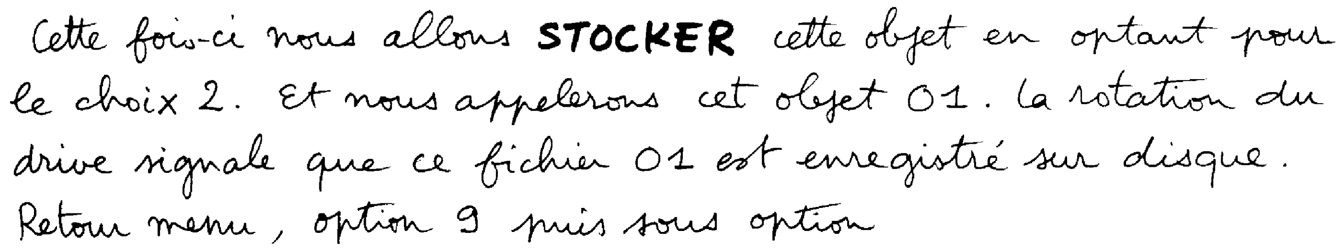
Ce carré va être entré sous forme d'une chaine constituée de 5 points

N°POINT	X	Y	Z
1	2	0	-4
2	9	0	-4
3	9	0	4
4	2	0	4
5	2	0	-4

Pas d'erreur, retour au menu.
Nous appelons alors l'option 9 - MANIPULER UN OBJET
Puis la sous option 2 - OPERER UNE ROTATION.
Apparait :

L'OBJET EST-IL EN MEMOIRE ?

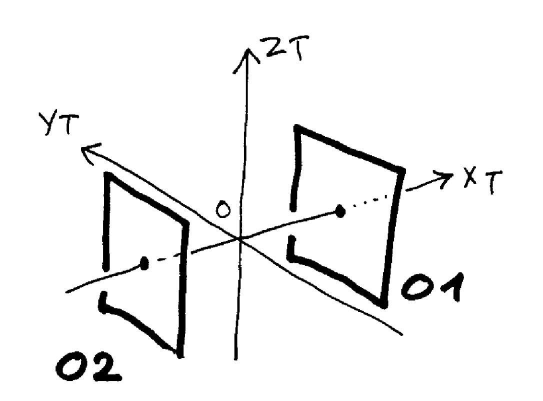
Réponse : oui
Nous allons alors choisir une rotation selon un axe parallèle à l'axe OZ (choix 3).
l'idée est de faire tourner ce carré comme ceci de 90 degrés.
la trace de l'axe dans le plan XOY correspond à XA = 6 YA = ∅
ANGLE, DEGRES : 9∅
Retour menu principal.



Cette fois-ci nous allons **STOCKER** cette objet en optant pour le choix 2. Et nous appelerons cet objet O1. La rotation du drive signale que ce fichier O1 est enregistré sur disque.
Retour menu, option 9 puis sous option

3 - CREER LE SYMETRIQUE % UN PLAN

l'objet O1 est toujours en mémoire.
Nous choisissons le plan YOZ, ce qui va substituer, en mémoire l'objet O2, symétrique de O1 :



Nous stockons ce nouvel objet O2.

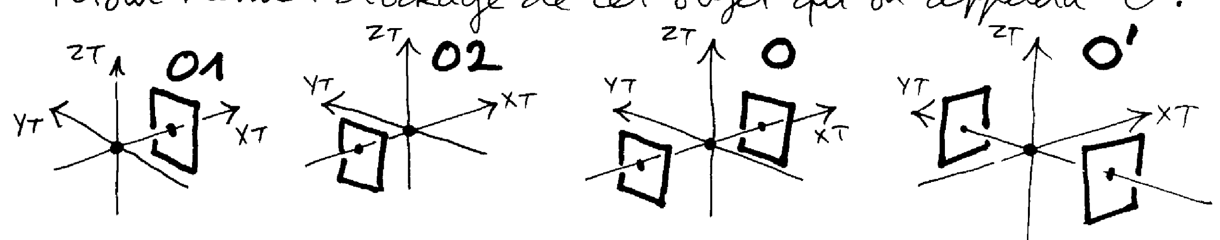
Retour menu. Choix 9
Et sous option
4 - FUSIONNER OBJETS

O2 est toujours en mémoire.

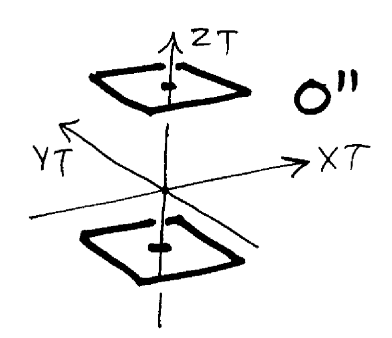
La machine demande qu'on lui indique le nom de l'objet à fusionner. Nous choisirons O1.
Nous pourrions ainsi fusionner toute une serie d'objets. Pour interrompre cette opération on tapera le mot FIN et nous appelerons l'objet global O
cet objet est donc la réunion de O1 et O2.
Retour menu, option 9, sous option

2 - OPERER UNE ROTATION

l'objet est en mémoire et nous provoquons une rotation de 90° autour de l'axe OZ (sous-sous option 3, XA = ∅, YA = ∅)
Retour Menu. Stockage de cet objet qu'on appelera O'.



Retour menu. Option 9. Sous option. L'objet O' est toujours en mémoire. Nous choisissons une rotation autour de l'axe OXT (YA = ∅, ZA = ∅, ANGLE, DEGRES : 9∅)
On stocke cet objet **O"**.

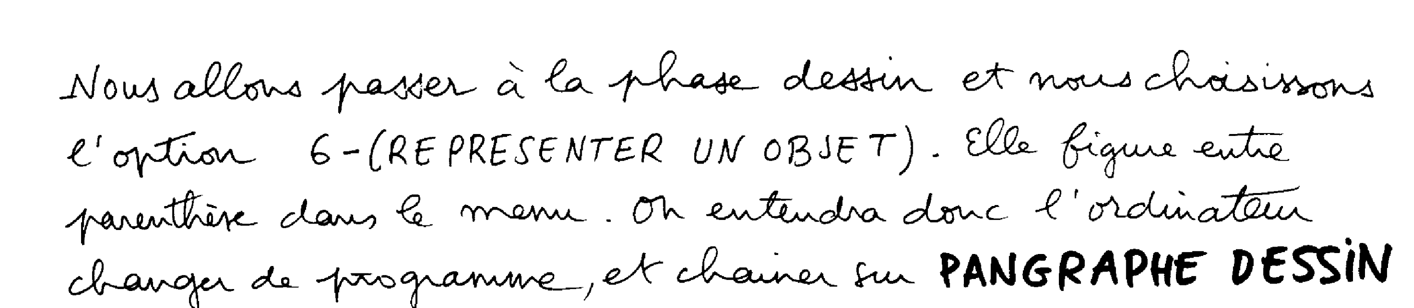


Retour menu. Option 9
sous option 4 (fusion). O" est toujours en mémoire.
On fusionne avec O' et O
et on nomme cet objet global **CUBE(∅/∅/∅)**
Entre parenthèses : les coordonnées du centre de gravité de cet objet.

Sur la disquette nous avons stocké un objet appelé

TRIEDRE(Ø/Ø/Ø)

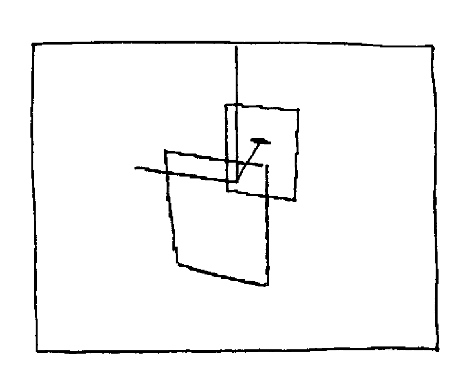
Une flèche indique l'axe OXT.



Nous allons passer à la phase dessin et nous choisissons l'option 6-(REPRESENTER UN OBJET). Elle figure entre parenthèse dans le menu. On entendra donc l'ordinateur changer de programme, et chainer sur **PANGRAPHE DESSIN**

OUVERTURE ANGULAIRE : 2Ø
POINT D'OBSERVATION
X = -25
Y = -8
Z = 12
CENTRE DE L'IMAGE
XG = Ø
YG = Ø
ZG = Ø

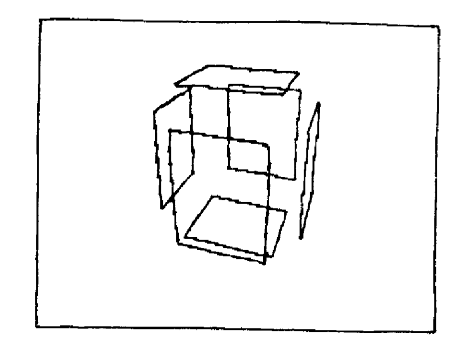
L'objet **O** est en principe toujours en mémoire. Ce qui fait qu'en choisissant l'option
1-OBJET DEJA EN MÉMOIRE
Nous voyons se dessiner :



Nous pouvons stocker cette image sur disque, et la rappeler, à volonté.

Retour menu Exercice : en gardant les mêmes paramètres de vision, tracez, par éléments
TRIEDRE(Ø/Ø/Ø)
O1
O2

on obtient l'image suivante :



Refaire l'opération en PAS à PAS.

Supposons que les objets **O**, **O'**, **O"** soient trop importants pour pouvoir être fusionnés (ce qui n'est pas le cas).
Nous allons, sur le Menu, choisir l'option

1Ø - DEFINIR UN BLOC D'OBJETS

NOMBRE D'OBJETS : 3
OBJET NO 1
O
OBJET NO 2
O'
OBJET NO 3
O"
NOM DE CE BLOC BLOC CUBE(Ø/Ø/Ø)

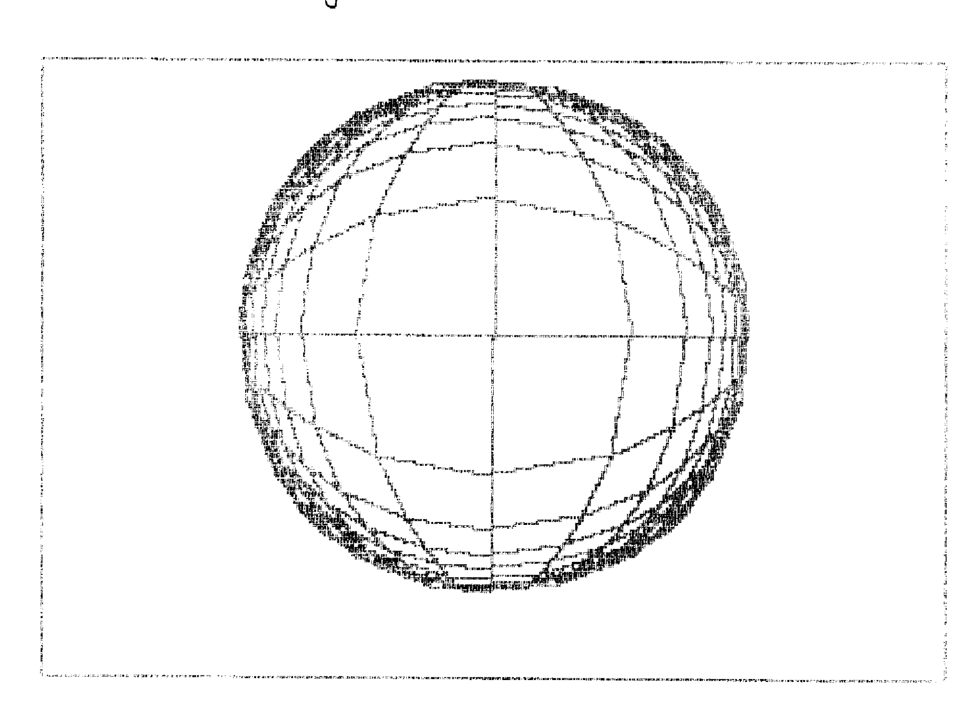
Nous pouvons alors envisager un dessin **PAR BLOC**
Passons (option 6) en Pangraphe dessin
mêmes paramètres de vision. Et nous choisissons la sous option 4 - DESSIN PAR BLOC
les trois objets du bloc sont gérés automatiquement.

LAVISION TOTALE

L'oeil humain possède un certain **CHAMP VISUEL**. Certains animaux ont des champs visuels très supérieurs, par exemple les poissons. Ce qui a donné lieu à des objectifs photographiques correspondant à des ouvertures angulaires de 90° (FISH-EYE ou oeil de poisson).

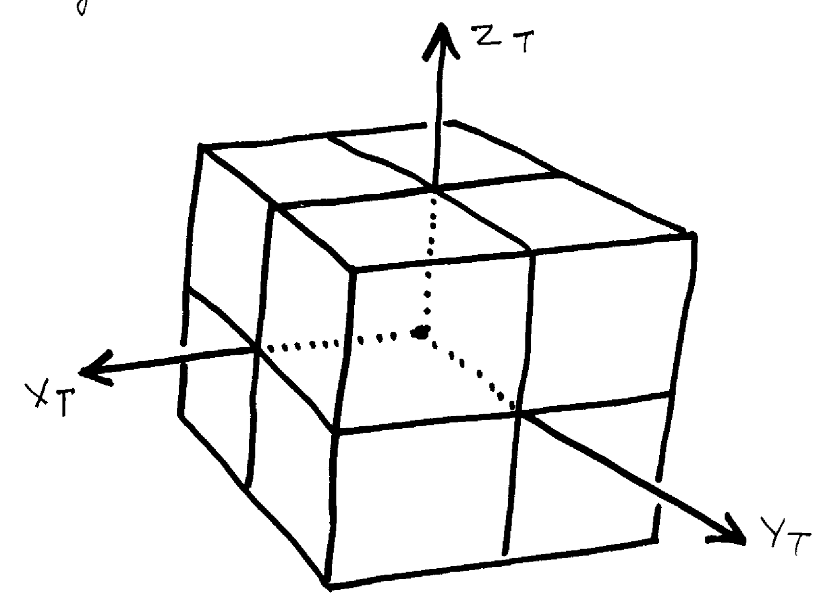
Si on a en mémoire un bloc grille composé d'éléments GRILLE1, GRILLE2, GRILLE3, GRILLE4, situés par exemple dans le plan Z = Ø on pourra observer cette **MIRE** en situant l'observateur sur l'axe OZT de telle manière que le point visé soit en (Ø, Ø, Ø).

Voici l'image de cette mire, vue à une "altitude" Z = 1 et avec une ouverture angulaire AN = 90°

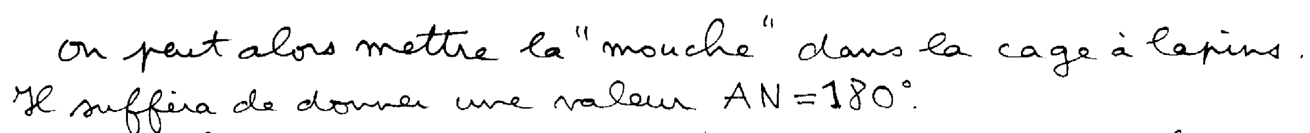


On retrouve l'aspect caractéristique des images FISH-EYE.

Ce qu'on pourra faire, maintenant, c'est construire un bloc suggérant une sorte de cage à lapins cubique, faite de grillage à maille carrée.



Il faudra partir de l'ensemble GRILLE 1, GRILLE 2, GRILLE 3, GRILLE 4, situé comme ceci :
Et utiliser les options rotation et translation.
Chaque face du cube sera constituée de 4 éléments, et le cube complet aura $6 \times 4 = 24$ éléments.

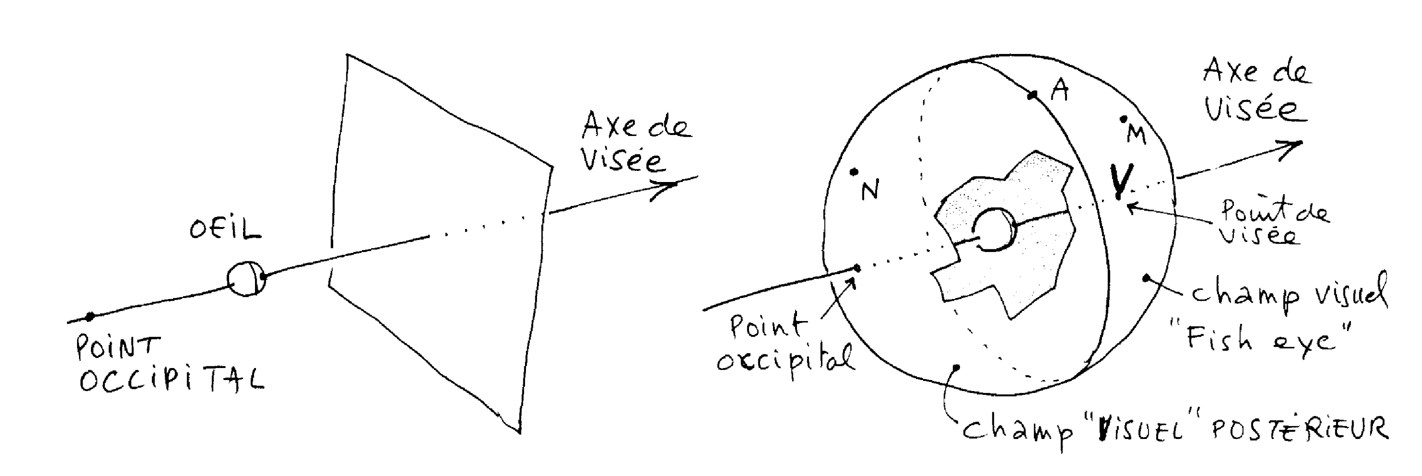


On peut alors mettre la "mouche" dans la cage à lapins.
Il suffira de donner une valeur AN = 180°.

La mouche pourra voir tous les éléments de la cage à lapin d'un seul coup d'oeil.

Sur l'écran, on pourrait tracer deux cercles concentriques.
Le centre de l'écran correspond à l'axe de visée.
Un premier cercle, de rayon 42.5 correspond à AN = 90°.
C'est la limite du champ de vision FISHEYE.

Un deuxième cercle, de rayon 85, correspond au **POINT OCCIPITAL**, c'est à dire situé sur la direction diamétralement opposée à l'**AXE DE VISÉE**.



champ visuel antérieur

champ visuel postérieur

Ecran graphique

A

N

V

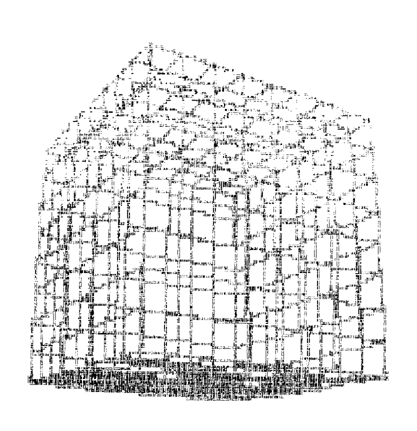
M

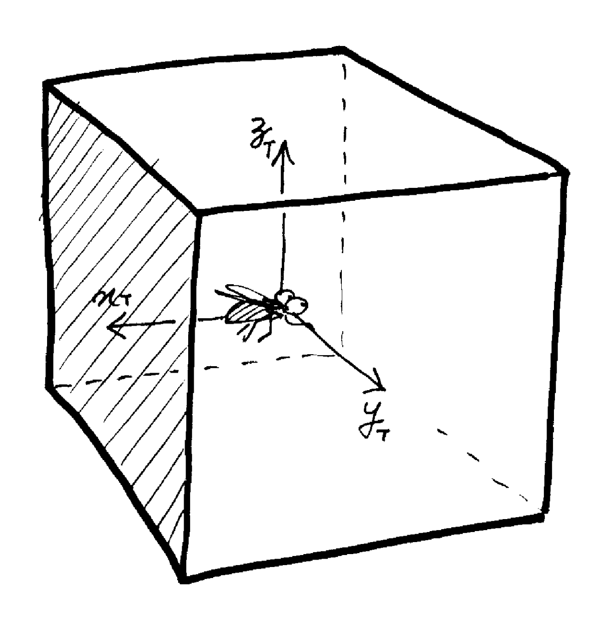
Image du point de Visée

Frontière du champ visuel de type fish eye

Cercle image du point OCCIPITAL

Sur la page suivant, la "CAGE"
au centre de laquelle on va placer la





Il suffira de poser la mouche par exemple au point

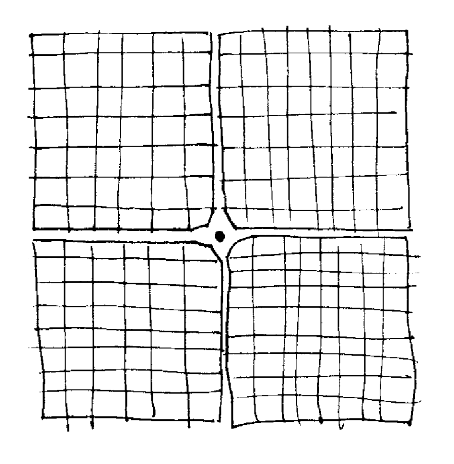
(∅.1 ; ∅ ; ∅)

et de la faire regarder dans la direction du point (∅ ; ∅ ; ∅)

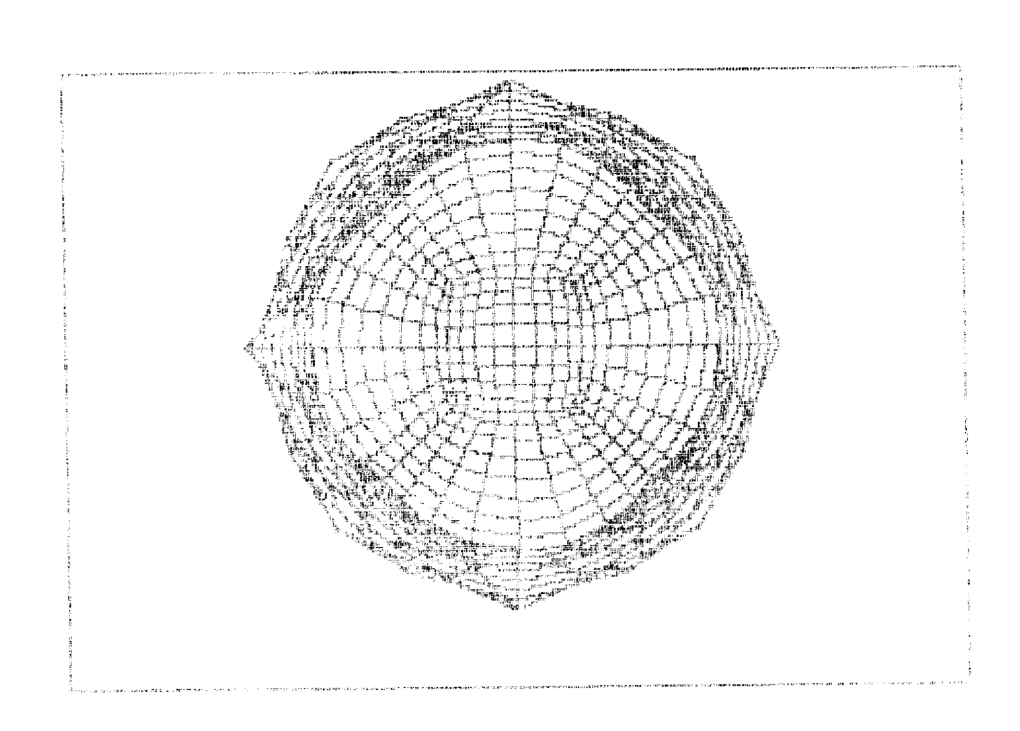
Valeur de AN = 180°.

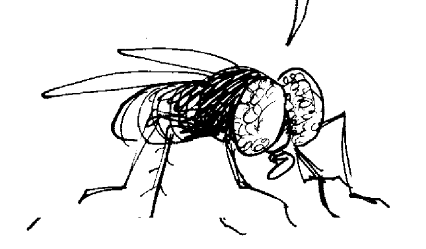
Il faudra prendre la précaution de ne pas matérialiser le point occipital.

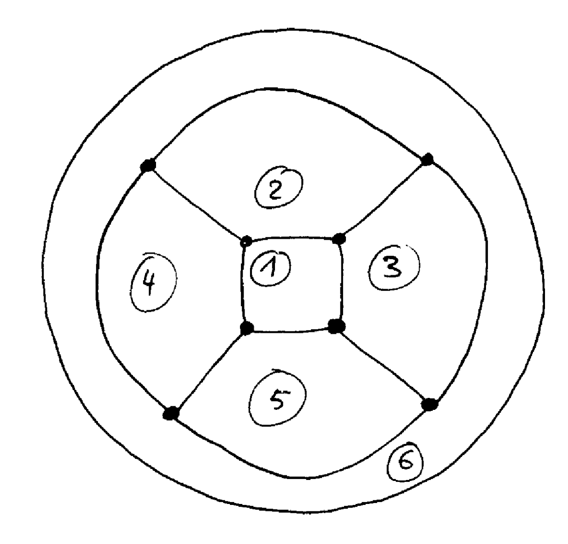
Voici la structure de la face hachurée faisant partie de la CAGE. Elle est formée de quatre éléments "écornés" (voir page **50**). Ainsi aucun des points de ce maillage n'appartient à l'axe OXT.

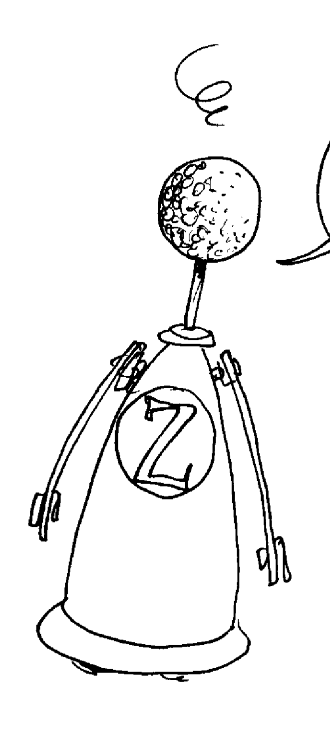


Dans ces condition, voila l'image de la cage cubique vu à travers cet **ŒIL DE MOUCHE.**



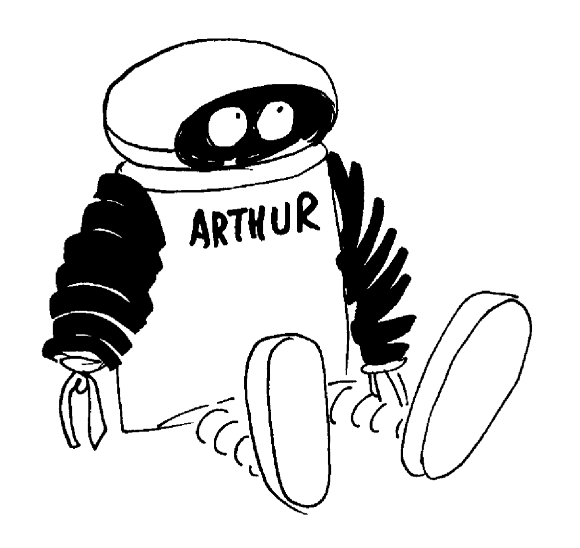






Bon, maintenant nous savons comment voient les mouches, les poissons, et même les hommes...

– Nous avons compris comment ils codaient dans leur tête le monde sensible, en fabriquant ces images sur leur rétine.
– les tableaux du musée sont donc la concrétisation de ce que les hommes voyaient dans leur tête.
– Faut croire...
– Bon, qu'est-ce que c'est que ce truc là-bas ?
– L'étiquette dit que c'est l'angélus de Millet.
– Tu sens quelque chose ?
– Rien. au point de vue émotion artistique, c'est raté...
– Peut-être qu'avec un 128 K ?...



FIN